Bioinformatics, 2023, **39(12)**, btad725
https://doi.org/10.1093/bioinformatics/btad725
Advance Access Publication Date: 29 November 2023
Original Paper

OXFORD

Genome analysis

ROCCO: a robust method for detection of open chromatin via convex optimization

Nolan H. Hamilton [1] **and Terrence S. Furey** [1,2,*]

[1]Department of Genetics, University of North Carolina at Chapel Hill, Chapel Hill, NC 27599, USA
[2]Department of Biology, University of North Carolina at Chapel Hill, Chapel Hill, NC, 27599, USA

*Corresponding author. Department of Genetics, University of North Carolina at Chapel Hill, 120 Mason Farm Road, CB#7264, Genetic Medicine Building, Chapel Hill, NC 27599, USA. E-mail: tsfurey@email.unc.edu (T.S.F.)

Associate Editor: Tobias Marschall

Abstract

Motivation: Analysis of open chromatin regions across multiple samples from two or more distinct conditions can determine altered gene regulatory patterns associated with biological phenotypes and complex traits. The ATAC-seq assay allows for tractable genome-wide open chromatin profiling of large numbers of samples. Stable, broadly applicable genomic annotations of open chromatin regions are not available. Thus, most studies first identify open regions using peak calling methods for each sample independently. These are then heuristically combined to obtain a consensus peak set. Reconciling sample-specific peak results *post hoc* from larger cohorts is particularly challenging, and informative spatial features specific to open chromatin signals are not leveraged effectively.

Results: We propose a novel method, ROCCO, that determines consensus open chromatin regions across multiple samples simultaneously. ROCCO employs robust summary statistics and solves a constrained optimization problem formulated to account for both enrichment and spatial dependence of open chromatin signal data. We show this formulation admits attractive theoretical and conceptual properties as well as superior empirical performance compared to current methodology.

Availability and implementation: Source code, documentation, and usage demos for ROCCO are available on GitHub at: https://github.com/nolan-h-hamilton/ROCCO. ROCCO can also be installed as a stand-alone binary utility using pip/PyPI.

1 Introduction

Nucleosomes, complexes of DNA and histone proteins, comprise the initial stage of chromatin compaction of the genome, reducing its occupying volume and enabling it to fit in cell nuclei (Li and Reinberg 2011). Most nucleosomal DNA is inaccessible for binding by transcription factors (TFs) that regulate gene expression. Genome-wide annotations of non-nucleosomal DNA, or open chromatin, therefore delineate where TFs can readily bind and effectively characterize the current gene regulatory program in a sample. Open chromatin landscapes vary across cell types and conditions, including in disease (Corces and Granja 2018) reflecting cell and condition-specific gene regulation. To better understand this dynamic nature of gene regulation, the identification of open chromatin regions has become an important aspect of molecular studies of complex phenotypes.

Several assays have been developed for genome-wide measurement of open chromatin, including DNase-seq (Boyle *et al.* 2008) and ATAC-seq (Buenrostro *et al.* 2015). These assays generate DNA fragments enriched for open chromatin regions that are then sequenced using short-read sequencers. Resulting reads are aligned to a reference genome, and regions with an enrichment of reads, or "peaks," are identified as open chromatin. Peak calling is a necessary step as, unlike genes for which annotations are available for many species, there are not comprehensive, predefined standard databases of open chromatin regions.

Studies focused on determining changes in chromatin associated with differing cellular conditions or complex traits normally include many samples. For these studies, it is necessary to define a common set of open chromatin regions, or "consensus peaks," to facilitate comparisons across sample groups. Typically, consensus peaks are determined by first annotating peaks independently in each sample. Then, these sample-specific peaks are merged based on one of several heuristics including: (i) simply take the maximal set across all samples. This method, though, is particularly vulnerable to anomalous data since peaks from a single sample satisfy the inclusion criterion; (ii) include only peak regions that occur in "all" samples. This is usually too stringent due to variability in data quality across samples, especially when there is an expectation of differences; and (iii) require that peaks be present in at least $M = 1 \ldots K$ samples, where the boundaries of the consensus peaks allow for some tolerance, T, for disparity in nucleotide position. Protocols in the spirit of this general method have been utilized in many open chromatin studies (Bao *et al.* 2015, Wang *et al.* 2018, Bentsen *et al.* 2020, Ming *et al.* 2021). A difficulty in applying such methods is choosing appropriate M and T—a task manifesting rigid criteria that may ignore some open regions or may include spurious regions and that may not define well-supported peak boundaries.

More statistically sound methods have been developed for handling multiple samples. For the specific case of $K = 2$

Received: 31 May 2023; **Revised:** 10 November 2023; **Editorial Decision:** 23 November 2023; **Accepted:** 28 November 2023

samples, the Irreproducible Discovery Rate (Li *et al.* 2011) can be controlled to mitigate calling of irreproducible peaks. However, since most experimental designs include $K \gg 2$ samples per group, it is difficult to apply this method broadly. Alternatively, Genrich (available at https://github.com/jsh58/Genrich) offers a method for multiple samples in which *P*-values are combined using Fisher's method (Fisher 1925). While Genrich has been used successfully in several studies (Hofvander *et al.* 2019, Guerin *et al.* 2021, Salavati *et al.* 2021, Tsaryk *et al.* 2022), the independence assumption of Fisher's method may be problematic for large numbers of samples and/or in cases involving multiple technical replicates (Roy *et al.* 2019). It also does not explicitly account for specific peak boundaries.

Here, we propose a novel method for identification of open chromatin regions across multiple samples, ROCCO: "Robust Open Chromatin detection via Convex Optimization." This method offers several favorable features:

- Accounts for both enrichment "and" spatial characteristics of open chromatin signals, the latter of which is an informative but often ignored aspect of ATAC-seq data that can be used to not only better detect regions but also improves on annotating peak boundaries;
- Leverages data from multiple samples without imposing arbitrary "hard" thresholds on a minimum number of samples declaring peaks;
- Is efficient for large numbers of samples;
- Does not require training data or a heuristically determined set of initial candidate regions, which are hard to define given the lack of a priori sets of validated open chromatin regions;
- Employs a mathematically tractable model granting useful performance guarantees.

We formally describe the algorithm utilized by ROCCO for the consensus peak problem and present a theoretical analysis. We then conduct several experiments to investigate ROCCO's efficacy empirically, using a set of 56 samples from human lymphoblastoid cell lines.

2 Materials and Methods

We begin by introducing notation (See Table 1 for a complete notation reference) used throughout this manuscript and describing the structure of signal data used by ROCCO to detect accessible chromatin.

2.1 Notation and definitions

Let $\mathscr{L}$ be a contiguous genomic sequence, e.g. a chromosome, divided into n fixed-width loci, each consisting of L nucleotides as in Fig. 1. For each sample j, we assume access to a signal, s_{i_j}, computed as a function of observed enrichment based on sequence reads at the ith locus. For K total samples, this yields a $K \times n$ signal matrix used as input to ROCCO:

$$\mathbf{S}_{\mathscr{L}} = \begin{pmatrix} s_{1_1} & s_{2_1} & \dots & s_{n_1} \\ s_{1_2} & s_{2_2} & \dots & s_{n_2} \\ \dots & \dots & \dots & \dots \\ s_{1_k} & s_{2_k} & \dots & s_{n_K} \end{pmatrix} = (\,\mathbf{s}_1 \quad \mathbf{s}_2 \quad \dots \quad \mathbf{s}_n\,), \qquad (1)$$

with $\mathbf{s}_i \in \mathbb{R}^K$ denoting the column vector of signal values among samples at the ith locus. This matrix can be generated

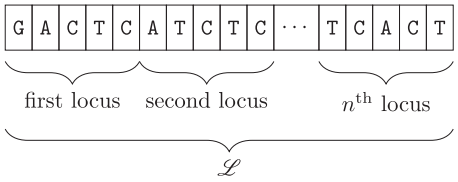

Figure 1. Genomic region $\mathscr{L}$ consisting of n fixed-width loci containing $L = 5$ nucleotides each.

with a variety of methods, but a context-specific tool, `rocco prep` is included as a subcommand in the software implementation for convenience: given a directory of samples' BAM files, $\mathbf{S}_{\mathscr{L}}$ is generated with multiple calls to PEPATAC's (Smith *et al.* 2021) `bamSitesToWig.py` tool.

2.2 Scoring loci

To determine consensus open chromatin regions, we first score each locus while accounting for enrichment (g_1), dispersion among samples (g_2), and a measure of local volatility in enrichment (g_3).

Specifically, we take $g_1(i)$ to be the median, and $g_2(i)$ to be the median absolute deviation (Pham-Gia and Hung 2001) of the K signal values at the ith locus:

$$g_1(i) = \text{med}\{s_{i_1}, s_{i_2}, \dots, s_{i_K}\}$$
$$g_2(i) = \text{med}\{|s_{i_1} - g_1(i)|, \dots, |s_{i_K} - g_1(i)|\}.$$

Large g_1 and low g_2 correspond to regions of high enrichment with little dispersion among samples—a favorable combination of traits to emphasize when predicting accessibility. We also leverage the disparities between enrichment signal values at adjacent loci, normalized by the current locus's enrichment,

$$g_3(i) = \frac{1}{g_1(i)+1} \begin{cases} |g_1(i) - g_1(i+1)|, & i = 1 \\ \max\{|g_1(i) - g_1(i+1)|, |g_1(i) - g_1(i-1)|\}, & 1 < i < n \\ |g_1(i) - g_1(i-1)|, & i = n \end{cases}.$$

A fundamental aim of g_3 is to more precisely annotate the edges and immediately adjacent regions of peaks where signals may be low before or after an abrupt shift in enrichment characterizing the nearby peak. See Fig. 2 for a visual demonstration on an idealized, continuous enrichment signal.

In (2), we define the score piece-wise and take a simple linear combination of g_1, g_2, g_3—or set this score to 0 if median enrichment is below a defined threshold. This enrichment threshold can encourage sparsity that may have favorable implications for computational efficiency during optimization and mitigate consideration of regions that are unlikely accessible.

$$\mathscr{S}(i) = \begin{cases} c_1 g_1(i) - c_2 g_2(i) + c_3 g_3(i) & \text{if } g_1(i) \geq \tau \\ 0 & \text{if } g_1(i) < \tau \end{cases}, \qquad (2)$$

where $\tau \geq 0$ is the minimum enrichment threshold placed on the median signal, and $c_1, c_2, c_3 \geq 0$ are coefficients for each term. By default, each scoring term has the following weights: $c_1, c_2, c_3 = 1$, and $\tau = 0$. These defaults provide strong performance but can be modified in the software implementation of ROCCO to suit users' specific needs. For example, users desiring more conservative peak predictions may wish to set $\tau > 0$.

$\mathscr{S}(i) \leq 0$ does not necessarily preclude the corresponding locus from being selected, since the objective function [see

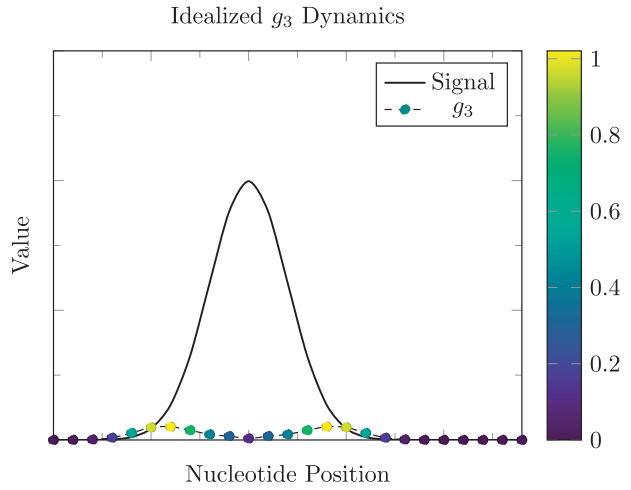

Figure 2. The g_3 term in Equation (2) is designed to more accurately determine peak edges to capture enriched regions in their whole. This function is greatest (yellow marks) near the ends of the enriched region and lowest (dark blue marks) at peak centers in this idealized example.

Equation (3)] is not completely dependent on the locus score. Note that for default parameters, $\mathscr{S}(i) \geq 0$.

2.3 Optimization

We address open chromatin detection as a constrained optimization problem. Let $\boldsymbol{\ell} \in \{0,1\}^n$ be a vector of binary decision variables, where we label $\ell_i = 1$ if the ith locus is present in open chromatin, and $\ell_i = 0$ otherwise.

We impose a budget constraint "upper-bounding" the proportion of selected loci in a given input chromosome. Let $b \in [0,1]$ be the maximum proportion of loci that can be selected, i.e.

$$\sum_{i=1}^{n} \ell_i \leq \lfloor nb \rfloor.$$

This constraint controls sensitivity in peak predictions and prevents unrealistic solutions in which an excessive fraction of chromatin is declared open. With estimates for the fraction of accessible chromatin hovering around $3\% - 4\%$ of the human genome (Song *et al.* 2011, Sahinyan *et al.* 2022), we accordingly set $b = 0.035$ as the default value. Since the budget applies to each chromosome independently, and the accessibility for each chromosome can vary, the software implementation allows for chromosome-specific parameters to be defined.

To optimize selection of accessible regions, the following objective function is minimized:

$$f(\boldsymbol{\ell}) = \underbrace{\sum_{i=1}^{n} -\Big(\mathscr{S}(i) \cdot \ell_i\Big)}_{\text{reward high locus scores}} + \underbrace{\gamma \sum_{i=1}^{n-1} |\ell_i - \ell_{i+1}|}_{\gamma \text{ controls influence of adjacent loci}}. \quad (3)$$

The first term rewards loci with high $\mathscr{S}$ scores, e.g. those with consistently high enrichment across samples or those on the edges of greatly enriched regions. The second term is introduced to account for spatial proximity of loci during optimization and controls the influence of signals in adjacent loci: for a given budget b, as γ is increased, fewer but longer distinct regions are annotated as open, yielding simpler solutions in a topological sense. This pattern is exhibited in Fig. 3.

To incorporate the described objective and budget constraint, we pose the following constrained optimization problem:

$$\begin{aligned} \underset{\boldsymbol{\ell}}{\text{Minimize}}: \quad & f_{\text{IP}}(\boldsymbol{\ell}) = \sum_{i=1}^{n} -\Big(\mathscr{S}(i) \cdot \ell_i\Big) + \gamma \sum_{i=1}^{n-1} |\ell_i - \ell_{i+1}| \\ \text{Subject To}: \quad & \text{(i)} \ \sum_{i=1}^{n} \ell_i \leq \lfloor nb \rfloor \\ & \text{(ii)} \ \ell_i \in \{0,1\}, \ \forall i = 1 \ldots n. \end{aligned} \quad (4)$$

Constraint (ii) restricts the feasible region to integer solutions. In general, such constraints yield difficult optimization problems, e.g. because gradients are not defined for functions over the integers. Indeed, general integer programing is known to be NP-hard (Korte and Vygen 2012). A common remedy is to convert the original, integer-constrained formulation to an analogous problem with convenient analytic properties. Accordingly, we substitute the constraints

$$\ell_i \in \{0,1\} \rightarrow \ \ell_i \in \mathbb{R} : 0 \leq \ell_i \leq 1$$

to obtain the following convex optimization problem:

$$\begin{aligned} \underset{\boldsymbol{\ell}}{\text{Minimize}}: \quad & f_{\text{CP}}(\boldsymbol{\ell}) = \sum_{i=1}^{n} -\Big(\mathscr{S}(i) \cdot \ell_i\Big) + \gamma \sum_{i=1}^{n-1} |\ell_i - \ell_{i+1}| \\ \text{Subject To}: \quad & \text{(i)} \ \sum_{i=1}^{n} \ell_i \leq \lfloor nb \rfloor. \\ & \text{(ii)} \ \ell_i \in [0,1], \ \forall i = 1 \ldots n. \end{aligned} \quad (5)$$

As we will see, this formulation maintains the essence of the original problem in (4) and confers several useful properties. In general, convexity is a highly valued feature in constrained optimization as it ensures every local minimum is also a global minimum, thereby preventing instances of "premature" convergence to suboptimal solutions (Boyd and Vandenberghe 2004).

Theorem 1 *The problem in (5) can be solved in polynomial time for a globally optimal solution.*

Linear programs (LPs) are a special class of convex optimization problems in which both the objective and constraints are linear functions of the decision variables. Though general convex problems can often be solved efficiently in practice, there are certain intractable instances. In contrast, LPs can be solved in worst-case polynomial time with respect to the number of variables (Boyd and Vandenberghe 2004). Accordingly, the proof of Theorem 1, deferred to Supplementary Section S1.1), relies on showing that an optimal solution to (5), $\boldsymbol{\ell}^{CP} \in \mathbb{R}^n$, is obtained from the n-dimensional truncation of the optimal solution to the following LP:

$$\begin{aligned} \underset{\boldsymbol{\ell}}{\text{Minimize}}: \quad & f_{\text{LP}}(\boldsymbol{\ell}) = \sum_{i=1}^{n} -\Big(\mathscr{S}(i) \cdot \ell_i\Big) + \gamma \sum_{j=n+1}^{2n-1} \ell_j \\ \text{Subject To}: \quad & \text{(i)} \ \sum_{i=1}^{n} \ell_i \leq \lfloor nb \rfloor \\ & \text{(ii)} \ \ell_i \in [0,1], \ \forall i = 1 \ldots n \\ & \text{(iii)} \ \ell_j \geq -1 \cdot (\ell_i - \ell_{i+1}), \ \forall i < n, \ j = n+i \\ & \text{(iv)} \ \ell_j \geq +1 \cdot (\ell_i - \ell_{i+1}), \ \forall i < n, \ j = n+i \end{aligned}, \quad (6)$$

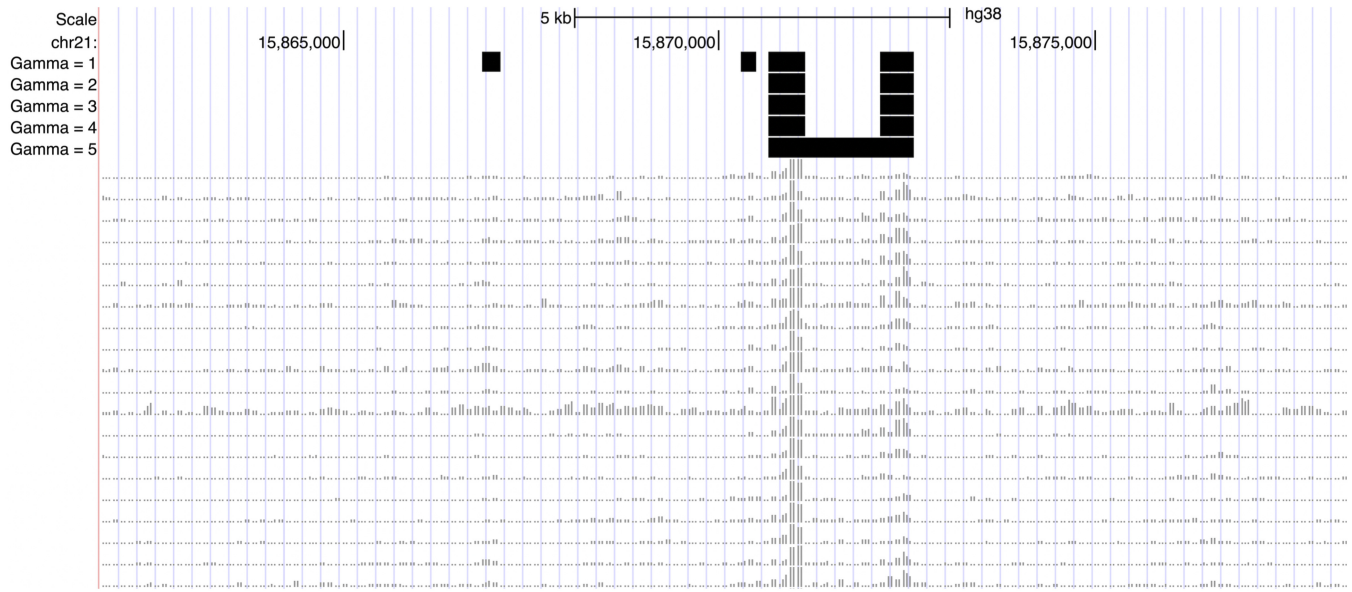

Figure 3. Example behavior of ROCCO in the UCSC Genome Browser (Karolchik *et al.* 2003) as γ is increased. The black bars in each track correspond to ROCCO's predictions given the samples' enrichment signals below. In the last row, we see that two distinct regions of enrichment are merged due to the strong influence of adjacent loci imposed by the $\gamma = 5$ parameter.

which we denote as $\boldsymbol{\ell}^{LP} \in \mathbb{R}^{2n-1}$. Moreover, the optimal objective values for (5) and (6) are equal:

$$\min f_{\text{CP}} = \min f_{\text{LP}} = \text{OPT}.$$

The time complexity of standard interior-point methods for solving LPs with n decision variables and a d-bit data representation is $\mathcal{O}(n^3 d)$, though several methods with improved worst-case bounds have been proposed (Karmarkar 1984, Vaidya 1989, den Hertog 1994). It is important to note that, in practice, modern solvers offer much greater efficiency than this worst-case bound might suggest by exploiting problem structure (Boyd and Vandenberghe 2004, Koch *et al.* 2022). Indeed, (6) possesses a particularly sparse objective and constraint matrix that allows for surprising efficiency showcased in Supplementary Material S1.3.

After solving the relaxed form of an integer program, it is often necessary to refine the solution for feasibility in the original integer-constrained region. Exact combinatorial techniques, such as branch and bound, can be applied to find optimal integer solutions; However, such methods may incur prohibitive computational expense and do not offer efficient runtime guarantees. It is often more practical to use an approximation scheme (Williamson and Shmoys 2011).

We find in our own experiments that solutions to (6) are nearly integral (Fig. 4) and can be rounded immediately, for instance, with the floor function, to obtain a feasible solution without a substantial sacrifice in practical performance. However, this near-integrality cannot be guaranteed in all experimental settings and parameter configurations, and a more robust procedure is preferred, which we now describe.

Given a solution to the relaxed formulation (6), we devise a procedure, denoted RR, based on randomized rounding (Raghavan and Tompson 1987), to obtain a set of candidate integral solutions, $\mathbf{L}_N$. This set is generated by executing N iterations of Algorithm 1, after which RR picks the best feasible solution with the lowest objective value.

Algorithm 1: Drawing $\boldsymbol{\ell}^{\text{rand}}$

Input: $\boldsymbol{\ell}^{LP}$
$\boldsymbol{\ell}^{rand} \leftarrow \mathbf{0}$
for $\ell_h \in \boldsymbol{\ell}^{rand}$ **do**
  Assign integer decision variable:
$$\ell_h^{rand} \sim \text{Bernoulli}(\ell_h^{LP})$$
end
Return: $\boldsymbol{\ell}^{rand}$

Note that in a given solution space $A \in \mathbb{R}^D, D \geq 1$, integral solutions cannot yield better performance than the best real-valued solution. The set of integral solutions is a proper subset of A. For this reason, the quality of integer solutions can be judged with reference to $f_{\text{LP}}(\boldsymbol{\ell}^{LP})$. In light of this, the construction of solutions $\boldsymbol{\ell}^{rand} \in \mathbf{L}_N$, with each ℓ_i^{rand} defined as a Bernoulli-distributed random variable with parameter $p_i = \ell_i^{LP}$, grants convenient properties arising from linearity of expectation. Namely,

$$\mathbb{E}[f_{\text{LP}}(\boldsymbol{\ell}^{rand})] = \text{OPT}$$

with constraints satisfied in expectation by $\boldsymbol{\ell}^{rand}$. We can use these expected values and leverage concentration inequalities to make probabilistic assertions regarding the solutions present in $\mathbf{L}_N$.

Theorem 2 *Let $\mathbf{L}_N$ be a set of $N \geq 1$ random solutions generated with Algorithm 1, and let $c > 1$, $a > 0$ be real numbers satisfying $\frac{1}{c} + e^{-2n(ab)^2} < 1$ for n loci and budget $b \in (0, 1)$. Then with high probability, $\mathbf{L}_N$ contains at least one solution with both (i) an objective*

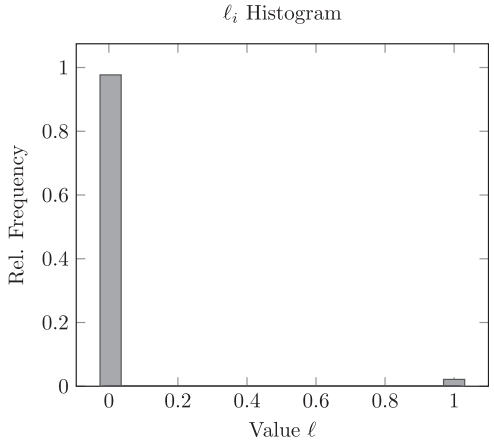

Figure 4. Observed distribution of decision variables after solving (6) with budgets $b \in \{.01, .025, .05, .075, .10\}$ on 50 random subsamples (K=40) of the ATAC-seq data detailed in "Data availability" section and pooling the solutions from each run.

value no more than $c \cdot \mathsf{OPT}$ *and (ii) no more than* $nb(1+a)$ *loci selected.*

In short, Theorem 2 is proven (Supplementary Material S1.2) using Markov's and Hoeffding's inequalities to show that the probability of one or more $\boldsymbol{\ell}^{rand} \in \mathbf{L}_N$ satisfying both criteria is at least

$$1 - \left(\frac{1}{c} + e^{-2n(ab)^2}\right)^N,$$

which quickly approaches 1 for increasing N. We emphasize that this expression is a lower bound on the probability of a satisfying solution, and we often observed multiple such solutions in $\mathbf{L}_N$ during the course of our experiments. But Theorem 2 allows us to make more general assertions under the supposition $\frac{1}{c} + e^{-2n(ab)^2} < 1$. Since n is on the order of millions for default locus size L=50, this criterion is satisfied even for quite small $c > 1, a > 0$.

The RR procedure has linear time and space complexity, making it a minor contributor to overall computational expense. Though this random procedure technically renders ROCCO stochastic, for the default N=50 RR iterations, we observed only minor variation in solutions returned from independent runs of ROCCO. As seen in Supplementary Material S1.5, a pairwise Jaccard similarity matrix for five independently generated ROCCO solutions contains values no less than 0.9977.

Algorithm 2 offers a pseudocode representation of ROCCO as a whole.

The object returned by Algorithm 2 is an n-dimensional decision vector, $\boldsymbol{\ell}^*$, used to select loci as accessible. Note, contiguous selections (i.e. sequences of loci such that $\ell_i^* = 1$) are merged into single peaks in the final BED file.

Remark 1 *Large Sample Sizes*. Note that only the scoring step is directly affected in runtime by the number of input samples K, running on the order of nK elementary operations using the median of medians algorithm. Practical scenarios satisfy $K \ll n$, making the number of input samples a minor contributor to computational expense during the calculation of $\mathscr{S}(\cdot)$. Further, because the scoring step is asymptotically dominated in worst-case computational expense by the optimization step, which is independent of K, the worst-case time complexity of ROCCO is likewise independent of the sample size, K.

Algorithm 2: ROCCO

Input: $\mathbf{S}_{chr} \in \mathbb{R}^{K \times n}$
Parameters: $b, \gamma, \tau, c_1, c_2, c_3$

1. *Scoring.* Compute (2):
   $\mathcal{S}(i) \leftarrow$ `score`$(\mathbf{S}_{chr}, \tau, c_1, c_2, c_3)$`[i]`
2. *Optimization.* Solve LP (6):
   $\boldsymbol{\ell}^{LP} \leftarrow$ `solve`$((6), b, \gamma)$;
3. *Solution Refinement.* Obtain integral solution from $\boldsymbol{\ell}^{LP}$:
   $\boldsymbol{\ell}^* \leftarrow$ `round_truncate`$(\boldsymbol{\ell}^{LP}$, `method=RR`$)$

Return: $\boldsymbol{\ell}^*$

3 Results

We performed several experiments to assess ROCCO's detection performance using ATAC-seq data from 56 human lymphoblast samples generated within the ENCODE project (See "Data availability" section). Experiments were conducted using a stand-alone computer with an Intel Xeon CPU E5-2680 v3 @ 2.50 GHz processor, 8 cores, and 64 g RAM. We ran the utility, `rocco prep`, included in the ROCCO software distribution, to process BAM files and create enrichment signal tracks with L=50. The MOSEK solver (https://mosek.com), for which a free academic license can be readily obtained, is used to solve the linear program (LP) in (6). Note, ROCCO can call any open-source solver offered within the CVXPY (Diamond and Boyd 2016) platform, but runtimes may vary. ECOS (Domahidi *et al.* 2013)is a viable option installed with CVXPY by default. Additional analyses and details are available in the Supplementary Material.

3.1 Detection performance

A noteworthy limitation in experiments comparing performance of open chromatin detection methods is a lack of high-confidence annotations against which to test. However, to gauge performance and ensure viability, some proxy for ground truth is needed. Following Zhao and Boyle (2021), we constructed a "union set," GT, of conservative irreproducible discovery rate-thresholded (Li *et al.* 2011) peaks from ENCODE "transcription factor" ChIP-seq experiments in the GM12878 lymphoblast cell line. We assume that the majority of annotated TF binding sites will correspond to open chromatin regions, but we note that variability in binding at a snapshot in time, the incomplete annotation of all TF binding, and cases where factors can bind to non-accessible chromatin introduce notable limitations. But, we argue that these data are sufficient to compare the relative performance of distinct methods. The $\mathscr{F}_\beta$-score, defined below, was then used to assess the ability to recover and bound regions in GT using ATAC-seq data from the 56 independent samples. Details regarding the construction of the GT dataset can be found in Supplementary Material S1.7.

For each method, we generated consensus peaks using previously determined alignments for the K=56 samples. We then computed precision as

$$\mathscr{P} = \frac{|D_X \cap D_{GT}|}{|D_X|}$$

and recall as

$$\mathscr{R} = \frac{|D_X \cap D_{GT}|}{|D_{GT}|,}$$

where D_X denotes the consensus peaks obtained from method X and set intersections in the numerators are computed using `bedtools intersect` (Quinlan and Hall 2010). The F_β-score was then calculated as the harmonic mean of precision and recall where recall is weighed β times as much as precision, i.e.

$$\mathscr{F}_\beta = (1+\beta^2)\frac{\mathscr{P}\cdot\mathscr{R}}{\beta^2\mathscr{P}+\mathscr{R}}.$$

As in Zhao and Boyle (2021), we use the $\mathscr{F}_\beta$-score as the primary metric for comparison of methods since it intuitively combines both precision and recall and is less affected by extreme regions of the precision–recall curve that do not correspond to realistic use-cases.

3.1.1 Detection performance: benchmark methods

Most methods to determine consensus peaks begin by identifying sample-specific peaks. For this step, we employed the widely used MACS2 software (Gaspar 2018) using parameters commonly specified for ATAC-seq experiments (see Section 5). With these, we used a common heuristic to specify consensus peaks (Yang *et al.* 2014). Namely, `MACS2-Consensus` only retained merged peaks supported by a majority of samples with a 100 bp tolerance in chromosome position across samples. Genrich is another method for consensus peak calling we tested that analyzes samples separately, calculating P-values for each. It then applies Fisher's method to combine P-values at each genomic region. We also generated peak sets with `MACS2-Pooled`, which combined alignments from all samples into one BAM file and then used MACS2 to call peaks on this combined alignment file.

See Supplementary Material S1.8 for exact configurations used to produce results for these MACS2-based methods and for Genrich.

3.1.2 Detection performance: results

For an initial visual comparison of methods, Fig. 5 displays peak calls from each in 100 and 20 kb regions on Chromosome 19 in the UCSC Genome Browser (Karolchik *et al.* 2003). We also include ATAC-seq signals from 25 of the lymphoblast samples being evaluated. As expected, all methods identify regions with consistently strong signals across all samples. They vary, though, in the contiguity and boundaries of these regions. There are also method-specific regions, as well as ones called by multiple, but not all, methods.

To quantify genome-wide detection performance of the methods, we evaluated across several values

$$\beta \in \{0.5, 0.75, \ldots, 1.50, 2.0\}$$

to address a plausible but encompassing range of recall/precision prioritizations. The most extreme cases $\beta = 0.5, \beta = 2.0$ were included for completeness but may not be particularly well-motivated by realistic usage since the corresponding $\mathscr{F}_\beta$ score can be unduly improved by simply rejecting any uncertain predictions or accepting all plausible predictions, respectively.

For each $\mathscr{F}_\beta$-score, we tuned each method over a range of significance thresholds deemed reasonable given their underlying models to maximize their performance. For `Genrich`, we tested

$$p \in \{10^{-6}, 10^{-5}, \ldots, 10^{-1}\}.$$

For the MACS2-based methods, we tested:

$$q \in \{.001, .005, .01, .05, .10, .20\}.$$

For `ROCCO`, the budget parameter is most fundamental and upper-bounds the fraction of genomic region $\mathscr{L}$ that can be selected. We thus use b as the tuning parameter for `ROCCO`, leaving $\gamma, \tau, c_{1,2,3}$ as their default values, and evaluated:

$$b \in \{.02, .025, \ldots, .06\}.$$

For each β listed above, given a method X and parameter r, we computed tuned performance as

$$\max_r \mathscr{F}_\beta(X_r),$$

i.e. the best-observed performance of the method while sweeping its most fundamental parameter. These values are recorded in Table 2. `ROCCO` matched or exceeded the best performance of every benchmark method for all six β values. The performance disparity between `ROCCO` and the second best-scoring method was smallest for the most recall-dependent case $\beta = 2$, which we have stated is only partially informative and particularly vulnerable to spurious predictions.

As mentioned above, `ROCCO` allows for specifying chromosome-specific parameters to account for varying chromatin state dynamics across the genome. For a cursory investigation into the effects of this practice, we tuned the budget parameter for $\mathscr{F}_1$ via grid search for each chromosome. We observed a non-trivial increase in performance ($\mathscr{F}_1 = .620$) compared to a constant budget for all chromosomes ($\mathscr{F}_1 = .579$ as in Table 2), but we expect additional improvements from a more rigorous, technically sound approach in which budgets are not restricted to an arbitrary set of values.

A comparison of methods using their "default" significance thresholds without tuning is included in Supplementary Material S1.4, where `ROCCO` offers the greatest $\mathscr{F}_\beta$-score in all but the $\beta = 0.5, \beta = 2.0$ experiments. Likewise, Supplementary Material S1.3 includes experiments assessing `ROCCO`'s computational efficiency in both theory and practice.

3.2 Variation in sample size/quality

Ideally, a consensus peak calling procedure will

- Effectively leverage data presented by multiple samples.
- Yield robust results in the presence of varying sample quality and size.
- Scale efficiently for large sample sizes.

In this section, we conduct several analyses to consider these aspects for `ROCCO`.

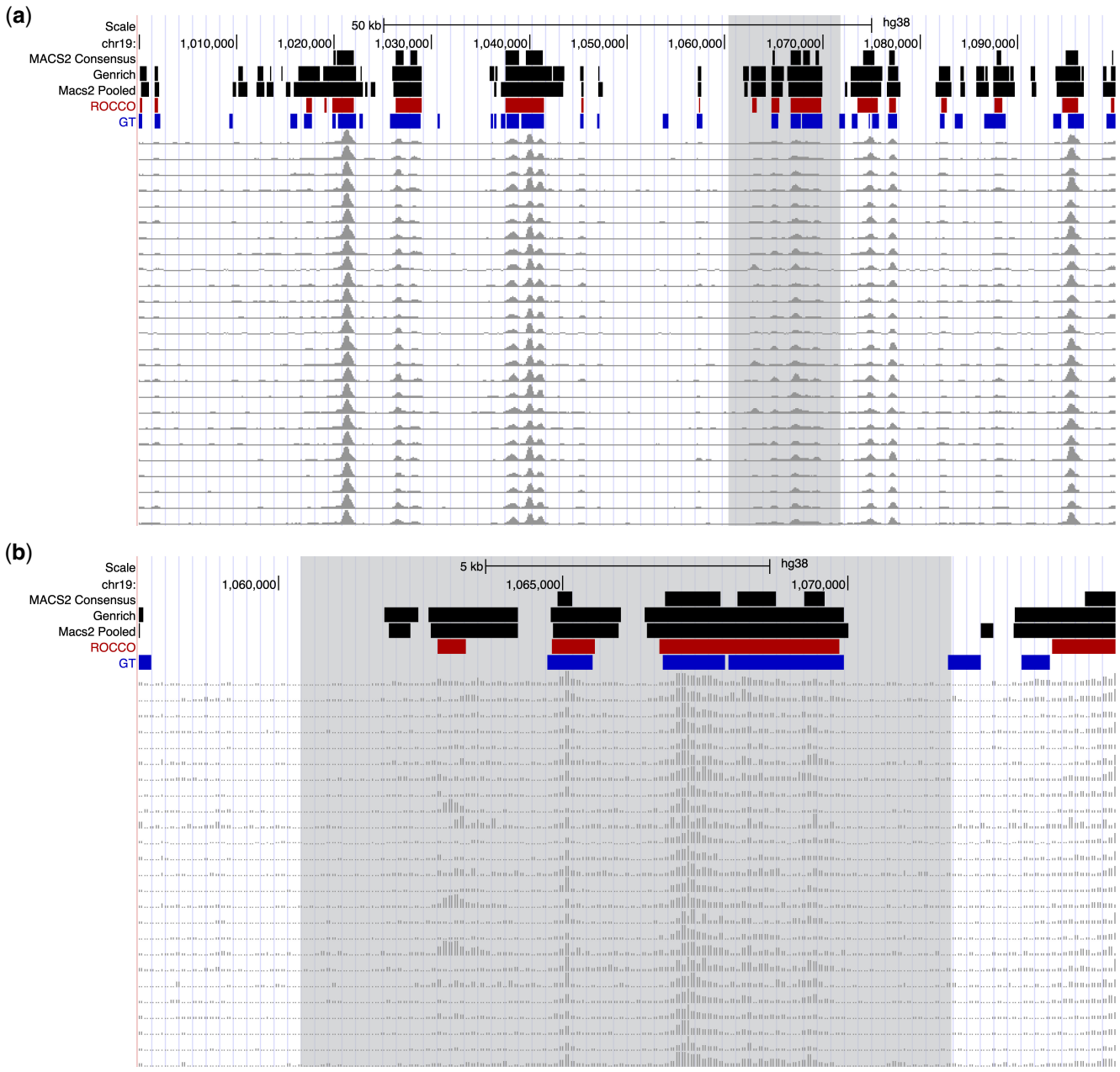

Figure 5. Example behavior over `chr19:1000000–1100000`. Consensus peak calls from each method tuned using the $\mathscr{F}_\beta$ score ($\beta = 1.0$). For perspective, results are displayed at two resolutions. (a) 100 `kb` and (b) 20 `kb`.

Table 1. Notation reference.

Symbol	Description	Default value
b	Budget threshold on selected loci	0.035
c_1, c_2, c_3	Weights for score function $\mathscr{S}(i)$	1.0
f	Objective function	N/A
γ	Fragmentation penalty in Equation (3)	1.0
K	Number of input samples	N/A
ℓ_i	Decision variable for ith locus	N/A
$\boldsymbol{\ell}$	Vector of decision variables	N/A
L	Fixed interval size of input signals	50
$\mathscr{L}$	Contiguous genomic region	N/A
n	Number of loci in $\mathscr{L}$	N/A
N	RR iterations	50
$\mathscr{S}(\cdot)$	Locus score function	N/A
$\mathbf{S}_\mathscr{L}$	$K \times n$ signal matrix over $\mathscr{L}$	N/A
τ	Median enrichment threshold in $\mathscr{S}(\cdot)$	0

Table 2. Performance for each method is recorded after tuning for the $\mathscr{F}_\beta$ value in the leftmost column.

β	ROCCO	Genrich	MACS2 − Pooled	MACS2 − Consensus
0.50	**0.651**	0.460	0.390	0.643
0.75	**0.596**	0.489	0.427	0.586
1.0	**0.579**	0.520	0.465	0.545
1.25	**0.580**	0.545	0.501	0.517
1.50	**0.582**	0.566	0.531	0.498
2.0	**0.603**	0.595	0.577	0.475

Bold values indicate the greatest score in each row across the tested methods.

In the first experiment, ROCCO is repeatedly executed using random subsamples of $K_{\text{sub}} \in \{5, 10, 15, \ldots, 50\}$ ATAC-seq alignments from the dataset in "Data availability" section as input. The subsamples' respective output peak sets are then

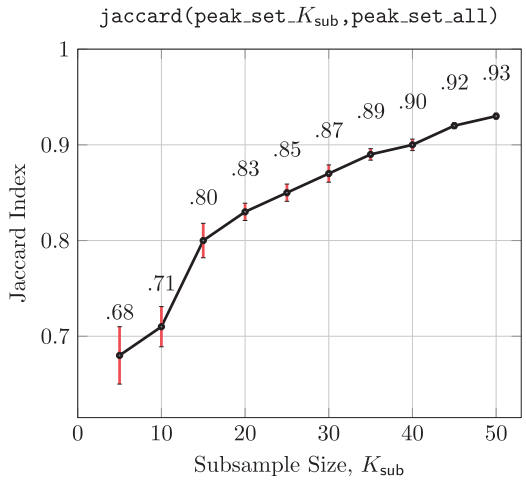

Figure 6. In 50 experiments for each $K_{sub} \in \{5, 10, 15, \ldots, 50\}$, K_{sub} ATAC-seq alignments are randomly subsampled and supplied as input to ROCCO. The 50 resulting output BED files are used to compute the average Jaccard index (95% CI) to ROCCO's results obtained using all $K=56$ samples.

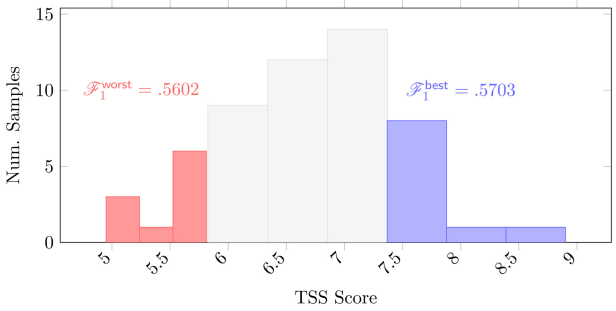

Figure 7. Histogram of TSS enrichment scores as defined by ENCODE for the $K=56$ ENCODE ATAC-seq samples used in experiments. TSS scores are commonly used as a quality measure for ATAC-seq alignments. ROCCO was run twice with default parameters—once using the $K=10$ "worst" samples (left/red) as input and again using the $K=10$ "best" samples (right/blue). The $\mathscr{F}_1$ performance in each case is labeled for comparison.

compared to the peak set obtained by running ROCCO on the full set of K=56 ATAC-seq alignments. To compute similarity between the subsamples' peak sets and the entire sample's, we measured the Jaccard statistic between their respective BED files using `bedtools` (Quinlan and Hall 2010). In this context, the Jaccard index measures the ratio of the number of intersecting base pairs to the number of base pairs in the union of two BED files. The average Jaccard index for each K_{sub} is recorded in Fig. 6 along with 95% confidence intervals. Notably, with only $K_{sub} = 5$ samples, ROCCO generated peak sets roughly 70%—similar to the ROCCO's peak set generated using all K=56 samples. Moreover, ROCCO produced strictly increasing Jaccard indexes for increasing subsample sizes, indicating an effective utilization of additional samples. Though the purpose of this experiment is to evaluate ROCCO's approximation of the full-sample-derived results with respect to smaller subsamples, we note that detection performance as measured in Section 3.1 likewise improved with respect to increasing K_{sub} from $\mathscr{F}_1 = 0.546 \pm 0.011$ to $\mathscr{F}_1 = 0.5783 \pm .001$.

Regarding efficiency, the cpu-time required to execute ROCCO genome-wide was affected negligibly by K_{sub}, with the average runtime for $K_{sub} = 5$ and $K_{sub} = 50$ differing by <20% despite the 1000% increase in samples. This result is informed theoretically by Remark 1, where the time complexity of ROCCO is shown to be asymptotically independent of sample size K.

The second experiment compares the effect of data quality on consensus peak sets generated by executing ROCCO independently on the ten best and worst samples as measured by the transcription start site (TSS) enrichment score (Smith *et al.* 2021). The data from the 56 lymphoblast samples are of relatively good quality (Supplementary Material S1.6), as evidenced by minimum TSS enrichment score of 4.95. Nonetheless, the distribution of scores reflects appreciable differences in sample quality between the left and right tails. With this considered, the relatively small disparities in ROCCO's detection performance shown in Fig. 7 indicate robustness to variation in sample quality. In comparison, `MACS2-Consensus`, the best-performing alternative method in this experimental setting, returns lower $\mathscr{F}_1$-scores for both the worst 10 samples ($\mathscr{F}_1 = 0.492$) and best 10 samples ($\mathscr{F}_1 = 0.530$) and a larger disparity between performance in each case.

3.3 Differential accessibility testing with ROCCO

A key motivation in the development of ROCCO was for experimental designs where ATAC-seq data are generated from multiple samples within two or more distinct groups. Using these data, a key question regards the location of genomic regions over which accessibility differs significantly between groups. Knowledge of such regions may yield insights into regulatory mechanisms responsible for phenotypic differences. To offer a template for differential analysis with ROCCO, we ran a simple experiment comparing the accessibility landscapes of males and females in the lymphoblast data described in "Data availability" section. Note, a Jupyter Notebook tutorial addresses steps for differential analysis with ROCCO and is available on the GitHub repository.

In this demonstrative experiment, ROCCO was run independently on the 23 female and 33 male lymphoblast samples using default parameters, and the peaks were merged *post hoc* to create a final set of 172 933 consensus peaks, which we refer to as `p-hoc_merge`. `p-hoc_merge` included 23 865 peaks only detected in the male samples and 19 165 peaks only detected in the female samples. ~17 000 peaks in each of these sets were not included in the set derived from running ROCCO on the input set of all K=56 samples, which we refer to as `all_k`. However, we note that the total span of disparate features was modest: a Jaccard similarity of ~85% was observed between `all_k` and `p-hoc_merge`. Whether to split by group and then merge or to run ROCCO on all samples combined is a context-specific decision dependent on parity in sample sizes/quality among the cohorts and the general motivation of the experiment. We accommodate both protocols and have made each straightforward to apply in ROCCO's software implementation.

Peak calling is an intermediate step in differential accessibility analyses to strategically identify candidate regions of interest, and DESeq2 (Love *et al.* 2014) was used in this experiment to detect significant differences in chromatin state between groups over the peak regions identified with ROCCO. At FDR-adjusted P<0.05, 3141 significant differentially accessible peaks spanning 2 275 100 bp were identified. About 93% (2916) of these peaks were observed in chromosome X, which is unsurprising given the recorded difference in sex between cohorts.

4 Discussion

In this manuscript, we introduced ROCCO, a novel method for identifying open chromatin regions in ATAC-seq data that simultaneously leverages information from multiple samples to determine a consensus set of peaks. ROCCO uses spatial features of enrichment signal data by initially formulating the problem with a convex model that can be solved with provable efficiency and performance guarantees. Importantly, the model accounts for features common to the edges of accessible chromatin regions, which are often hard to determine based on independently determined sample peaks that can vary widely in their genomic locations. In addition to several attractive conceptual and theoretical features, ROCCO also exhibited improved detection performance based on ATAC-seq data from 56 lymphoblast samples evaluated against known TF binding sites determined using ChIP-seq. ROCCO is especially suited for experimental designs that include multiple samples from two or more distinct groups with one goal being to determine regions that are differentially accessible between these groups. A Jupyter notebook tutorial provides a step-by-step protocol for this with all necessary scripts provided on the GitHub repository.

For simplicity and to provide a conservative comparison with other methods, we ran ROCCO with the same genome-wide parameters for all chromosomes, including the budget which dictates the "maximum" proportion of the chromosome that should be considered accessible. However, chromatin accessibility varies across chromosomes, and ROCCO's performance may be improved by exploiting properties specific to each chromosome. We found that optimizing the budget parameter for each chromosome for $\mathscr{F}_\beta$-score at $\beta = 1.0$ did show an improvement. These optimized budget parameters roughly reflected the differences in gene density and read density across chromosomes, as expected. In the future, we will focus on developing an efficient, robust method to derive reasonable chromosome-specific budgets based on the input signal data.

ROCCO's locus size parameter was set to L=50 throughout experiments. While our results suggest this grants good performance and a resolution sufficient to identify both broad and concentrated regions of enrichment, it may prove beneficial to modify this parameter depending on the expected size of elements and desired granularity. We note, however, that decreasing L increases the number of loci, n, which may induce additional computational expense. By the same reasoning, computational burden can be reduced by increasing L, though some loss in the precision of predicted peaks may result. The effects of the locus size parameter are discussed in greater detail in Supplementary Material S1.4.

Overall, ROCCO represents a scalable, effective, and mathematically sound method that is broadly applicable and addresses an important need in functional genomics analysis.

Supplementary data

Supplementary data are available at *Bioinformatics* online.

Conflict of interest

None declared.

Funding

This work was supported by the National Institutes of Health [P01DK094779, R01DK136262]; and the University of North Carolina BBSP Graduate Program.

Data availability

The ATAC-seq data from 56 lymphoblast samples used to conduct experiments were obtained from the ENCODE Project (Luo *et al.* 2019). Specifically, we used alignments that had been determined according to the ENCODE ATAC-seq protocol. Note, we remove chromosome Y from consideration to ensure each sample contained data for the same set of chromosomes. A link to the metadata with accession codes for this dataset is available in Supplementary Material S1.6.

References

Bao X, Rubin AJ, Qu K *et al.* A novel ATAC-seq approach reveals lineage-specific reinforcement of the open chromatin landscape via cooperation between BAF and p63. *Genome Biol* 2015;**16**:284.

Bentsen M, Goymann P, Schultheis H *et al.* ATAC-seq footprinting unravels kinetics of transcription factor binding during zygotic genome activation. *Nat Commun* 2020;**11**:4267.

Boyd S, Vandenberghe L. *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

Boyle AP, Davis S, Shulha HP *et al.* High-resolution mapping and characterization of open chromatin across the genome. *Cell* 2008;**132**:311–22.

Buenrostro JD, Wu B, Chang HY *et al.* ATAC-seq: a method for assaying chromatin accessibility genome-wide. *Curr Protoc Mol Biol* 2015;**109**:21.29.1–9.

Corces MR, Granja JM, Shams S *et al.*; Cancer Genome Atlas Analysis Network. The chromatin accessibility landscape of primary human cancers. *Science* 2018;**362**:eaav1898.

den Hertog D. *Interior Point Approach to Linear, Quadratic and Convex Programming*. Dordrecht, NL: Springer Netherlands, 1994.

Diamond S, Boyd S. CVXPY: a python-embedded modeling language for convex optimization. *J Mach Learn Res* 2016;**17**:1–5.

Domahidi A, Chu E, Boyd S. ECOS: an SOCP solver for embedded systems. In: *European Control Conference (ECC)*, Zurich, Switzerland. 3071–6. IEEE, 2013.

Fisher R. *Statistical Methods for Research Workers*. Edinburgh, UK: Oliver & Boyd, 1925.

Gaspar JM. Improved peak-calling with MACS2. bioRxiv, 2018, preprint: not peer reviewed.

Guerin LN, Barnett KR, Hodges E. Dual detection of chromatin accessibility and DNA methylation using ATAC-me. *Nat Protoc* 2021;**16**:5377–97.

Hofvander J, Puls F, Pillay N *et al.* Undifferentiated pleomorphic sarcomas with PRDM10 fusions have a distinct gene expression profile. *J Pathol* 2019;**249**:425–34.

Karmarkar N. A new polynomial-time algorithm for linear programming. In: *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*, STOC '84, Washington DC, USA. 302–11. New York, NY: Association for Computing Machinery, 1984.

Karolchik D, Baertsch R, Diekhans M *et al.*; University of California Santa Cruz. The UCSE Genome Browser Database. *Nucleic Acids Res* 2003;**31**:51–4.

Koch T, Berthold T, Pedersen J *et al.* Progress in mathematical programming solvers from 2001 to 2020. *EURO J Comput Optim* 2022;**10**:100031.

Korte B, Vygen J. *Combinatorial Optimization: Theory and Algorithms*. 5th edn. New York, New York: Springer Publishing Company, Incorporated, 2012.

Li G, Reinberg D. Chromatin higher-order structures and gene regulation. *Curr Opin Genet Dev* 2011;**21**:175–86.

Li Q, Brown JB, Huang H *et al.* Measuring reproducibility of high-throughput experiments. *Ann Appl Stat* 2011;**5**: 1752–79.

Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* 2014;**15**: 550.

Luo Y, Hitz BC, Gabdank I *et al.* New developments on the encyclopedia of DNA elements (ENCODE) data portal. *Nucleic Acids Res* 2019;**48**:D882–9.

Ming H, Sun J, Pasquariello R *et al.* The landscape of accessible chromatin in bovine oocytes and early embryos. *Epigenetics* 2021;**16**: 300–12.

Pham-Gia T, Hung T. The mean and median absolute deviations. *Math Comput Model* 2001;**34**:921–36.

Quinlan AR, Hall IM. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* 2010;**26**:841–2.

Raghavan P, Tompson CD. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica* 1987;**7**:365–74.

Roy A, Harrar SW, Konietschke F. The nonparametric Behrens-Fisher problem with dependent replicates. *Stat Med* 2019;**38**: 4939–62.

Sahinyan K, Blackburn DM, Simon M-M *et al.* Application of ATAC-Seq for genome-wide analysis of the chromatin state at single myofiber resolution. *eLife* 2022;**11**:e72792.

Salavati M, Woolley SA, Araya YC *et al.* Profiling of open chromatin in developing pig (*Sus scrofa*) muscle to identify regulatory regions. *G3 (Bethesda)* 2021;**12**:jkab424.

Smith JP, Corces MR, Xu J *et al.* PEPATAC: an optimized pipeline for ATAC-seq data analysis with serial alignments. *NAR Genom Bioinform* 2021;**3**:lqab101.

Song L, Zhang Z, Grasfeder LL *et al.* Open chromatin defined by DNaseI and FAIRE identifies regulatory elements that shape cell-type identity. *Genome Res* 2011;**21**:1757–67.

Tsaryk R, Yucel N, Leonard EV *et al.* Shear stress switches the association of endothelial enhancers from ETV/ETS to KLF transcription factor binding sites. *Sci Rep* 2022;**12**:4795.

Vaidya P. Speeding-up linear programming using fast matrix multiplication. In: *30th Annual Symposium on Foundations of Computer Science*, Durham, NC. IEEE, 1989.

Wang J, Zibetti C, Shang P *et al.* ATAC-seq analysis reveals a widespread decrease of chromatin accessibility in age-related macular degeneration. *Nat Commun* 2018;**9**:1364.

Williamson DP, Shmoys DB. *The Design of Approximation Algorithms*. Cambridge, UK: Cambridge University Press, 2011.

Yang Y, Fear J, Hu J *et al.* Leveraging biological replicates to improve analysis in ChIP-seq experiments. *Comput Struct Biotechnol J* 2014; **9**:e201401002.

Zhao N, Boyle AP. F-Seq2: improving the feature density based peak caller with dynamic statistics. *NAR Genom Bioinform* 2021;**3**:lqab012.