UNIVERSITY OF CALIFORNIA

Los Angeles

Statistical and Computational Methods for Comparing High-Throughput Data from Two Conditions

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Philosophy in Statistics

by

Xinzhou Ge

2021

ABSTRACT OF THE DISSERTATION

Statistical and Computational Methods for Comparing High-Throughput Data from Two Conditions

by

Xinzhou Ge
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2021
Professor Jingyi Jessica Li, Chair

The development of high-throughput biological technologies have enabled researchers to simultaneously perform analysis on thousands of features (e.g., genes, genomic regions, and proteins). The most common goal of analyzing high-throughput data is to contrast two conditions, to identify "interesting" features, whose values differ between two conditions. How to contrast the features from two conditions to extract useful information from high-throughput data, and how to ensure the reliability of identified features are two increasingly pressing challenge to statistical and computational science. This dissertation aim to address these two problems regarding analysing high-throughput data from two conditions.

My first project focuses on false discovery rate (FDR) control in high-throughput data analysis from two conditions. FDR is defined as the expected proportion of uninteresting features among the identified ones. It is the most widely-used criterion to ensure the reliability of the interesting features identified. Existing bioinformatics tools primarily control the FDR based on p-values. However, obtaining valid p-values relies on either reasonable assumptions of data distribution or large numbers of replicates under both conditions, two requirements that are often unmet in biological studies. In Chapter 2, we propose Clipper, a general statistical framework for FDR control without relying on p-values or specific data distributions. Clipper is applicable to identifying both enriched and differential features from high-throughput biological data of diverse types. In comprehensive simulation and real-data benchmarking, Clipper outperforms existing generic FDR control methods and specific bioinformatics tools designed for various tasks, including peak calling from ChIP-seq data, and differentially expressed gene identification from bulk or single-cell RNA-seq data. Our results demonstrate Clipper's flexibility and reliability for FDR control, as well as its broad applications in high-throughput data analysis.

My second project focuses on alignment of multi-track epigenomic signals from different samples or conditions. The availability of genome-wide epigenomic datasets enables in-depth studies of epigenetic modifications and their relationships with chromatin structures and gene expression. Various alignment tools have been developed to align nucleotide or protein sequences in order to identify structurally similar regions. However, there are currently no alignment methods specifically designed for comparing multi-track epigenomic signals and detecting common patterns that may explain functional or evolutionary similarities. We propose a new local alignment algorithm, EpiAlign, designed to compare chromatin state sequences learned from multi-track epigenomic signals and to identify locally aligned chromatin regions. EpiAlign is a dynamic programming algorithm that novelly incorporates varying lengths and frequencies of chromatin states. We demonstrate the efficacy of EpiAlign through extensive simulations and studies on the real data from the NIH Roadmap Epigenomics project. EpiAlign can also detect common chromatin state patterns across multiple epigenomes from conditions, and it will serve as a useful tool to group and distinguish epigenomic samples based on genome-wide or local chromatin state patterns.

The dissertation of Xinzhou Ge is approved.

Mark Stephen Handcock

Dinesh Rao

Hongquan Xu

Jingyi Jessica Li, Committee Chair

University of California, Los Angeles

2021

Dedicated to my family, in particular my mother Huiqin Jin, my father Xiaoheng Ge, and my girlfriend Huilin Chen.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

During my doctoral studies at UCLA, I received countless help from my advisor, collaborators, family, and friends.

First and foremost, I would like to express my eternal gratitude to my advisor, Dr. Jingyi Jessica Li, who has always supported my research and career development. Since I joined Jessica's group in Jan. 2017, when I was a Master student and not determined to seek an academic career, Jessica has led me into my current research field, supported my dream of academic career, and generously provided me with any help I could think about. I would also like to thank Jessica for treating us as her friends in daily life. We could discuss everything and shared our opinions towards the world and history during our lab activities, which will always be my treasured memories. Working with Jessica makes the journey of doctoral studies meaningful and rewarding.

I would like to thank Dr. Hongquan Xu, Dr. Mark Stephen Handcock, and Dr. Dinesh S. Rao for serving on my doctoral committee and for providing valuable feedback on my research. I want to thank Dr. Wei Li from UC Irvine and Dr. Leo Wang from City of Hope for their contributions to the first part of this dissertation.

I thank all current and former members of Jessica's group for their helpful discussions and suggestions about my research, particularly Dr. Wei Vivian Li, Dr. Yuchen Yang, Dongyuan Song, Tianyi Sun, Kexin Li, and Ruochen Jiang, for their help and comments on my research projects. Among them, I want to especially thank Yiling Chen, who is not only a collaborator, but also one of my best friends.

Finally, I would like to especially thank my family and friends for their love and support. My mother, Huiqin Jin, and my father, Xiaoheng Ge, have always supported me unconditionally. They have always believed in me and encouraged me to pursue my dream. I would like to thank my girlfriend, Huilin Chen, who has accompanied me through my whole doctoral journey. Her optimistic nature can always give me strength and power. I thank my longtime friends, Yian Yin, Zhengdong Ge, Yijun Shen, Tian Liu, Renfei Huang, and Chao Zhang (just to name a few) for always being there when I need them.

VITA

2012-2016	B.S. in Statistics, Peking University
2016-2021	Graduate Student Researcher, Department of Statistics, UCLA

PUBLICATIONS

(* indicates equal contribution.)

Ge, X*, Zhang, H*, Xie L, Li, W. V., Kwon, SB., & Li, J. J (2019). EpiAlign: an alignment-based bioinformatic tool for comparing chromatin state sequences. Nucleic acids research 47 (13), e77-e77.

Ge, X.*, Chen, Y. E.*, Song, D., McDermott, M., Woyshner, K., Manousopoulou, A., Li, W., Wang, L. D., & Li, J. J. (2020). Clipper: p-value-free FDR control on high-throughput data from two conditions. (manuscript)

Lyu, J., Li, J. J., Su, J., Peng, F., Chen, Y. E., **Ge, X.**, & Li, W. (2020). DORGE: Discovery of Oncogenes and tumoR suppressor genes using Genetic and Epigenetic features. Science advances, 6(46), eaba6784.

Li, S., Dou, X., Gao, R., **Ge, X.**, Qian, M., & Wan, L., (2018).A remark on copy number variation detection methods. PloS one 13 (4), e0196226.

CHAPTER 1

Introduction

The development of high-throughput technologies in the past decades has greatly revolutionized the field of molecule biology, by enabling biologists to measure system-wide biological features, such as genes, genomic regions, and proteins ("high-throughput" means the number of features is large, at least in thousands). These high-throughput technologies have led to important scientific discoveries [1–3], as well as new challenges for statistical and computational method development. Two important high-throughput technologies are RNA sequencing (RNA-seq), which allows for genome-wide profiling of transcriptome landscapes, and chromatin immunoprecipitation followed by sequencing (ChIP-seq), which captures genome-wide protein interactions with DNA.

The RNA-seq technology aims to capture RNA contents of a biological sample by indirectly sequencing cDNAs reversely transcribed from extracted RNAs. As RNA-seq technologies have greatly lowered the cost, as well as increased the coverage and accuracy of sequencing, measuring transcriptomes, which consist of RNA transcripts of all genes from an individual or a population of cells, by RNA-seq has become one of the most popular topics in genomics research. RNA-Seq can be used to quantify gene expression levels and identify novel genes/transcripts, alternative splicing events, and rare genetic variants in a biological sample. As transcriptomes vary across tissues and cell types, differential analysis using RNA-seq data measured under different conditions has shed insights into molecular functions and processes such as cellular differentiation, carcinogenesis, and transcription regulation.

ChIP-seq is a genome-wide experimental assay for measuring binding intensities of a DNA-associated protein [4], such as a transcription factor that activates or represses gene expression [5, 6]. Chromatin immunoprecipitation can isolate specific DNA sites in direct physical interaction with a protein of interest, e.g., a transcription factor, thus produces a library of target DNA sites bound to the protein. ChIP-seq data are crucial for studying gene expression regulation. An indispensable analysis, termed "peak calling," identifies genomic regions with enriched sequence reads in ChIP-seq data; these regions are likely bound by the target protein and thus of biological interest. As ChIP-seq can also reveal patterns of many epigenetic chromatin modifications, it has become a key experimental method used in epigenomic research. Genome-wide analysis of histone modifications, such as genome-wide annotation of chromatin states, has enabled systematic analysis of how the epigenomic landscape contributes to cell identity, cellular processes, gene expression and disease.

High-throughput datasets often contain biological features measured under more than one condition, for example, experimental versus control condition or different cell types. The most common goal of analyzing high-throughput data is to contrast two conditions so as to reliably screen "interesting features," which exhibit an elevated or differential measurement across conditions. Two typical such analyses are the identification of differentially expressed genes (DEGs) from genome-wide RNA-seq gene expression data, and calling protein-binding sites in a genome from chromatin immunoprecipitation sequencing (ChIP-seq) data. DEG analysis, where each feature is a gene, aim to identify genes whose expression levels change between two conditions. Peak calling from ChIP-seq data, where each feature is a genomic region, aim to identify genomic regions with enriched sequence reads in ChIP sample, in contrast to a negative control sample. The identified interesting features are called discoveries, and are subject to further investigation and validation. As the number of features in high-throughput data is tremendously large, researchers demand reliable discoveries that only contain few false discoveries to reduce experimental validation that is often laborious or expensive. Therefore, the false discovery control is a key problem in high-throughput data analysis comparing different conditions.

Another problem in high-throughput data analysis comparing different conditions is how to define a measurement to summarize comprehensive information from two conditions. One example for such problem is the comparison of multi-track epigenetic signals. Epigenome encodes information of chemical modifications to DNA and histone proteins in a genome, and such modifications may result in changes to chromatin structures and genome functions. Epigenomic information is represented by multi-track signals, including DNA methylation, covalent histone modifications, and DNA accessibility, all of which are measured genome-wide by high-throughput sequencing technologies such as ChIP-seq. The multi-track nature of epigenomic signals is a challenge for measuring the similarity or difference of a genomic region in different samples (e.g., under two conditions), or of two genomic regions in the same sample.

My dissertation will focus on the above two problems. For the false discovery control problem, we proposed Clipper, a p-value-free false discovery rate (FDR) control framework on high-throughput data from two conditions. For contrasting multi-track epigenetic signals, we proposed EpiAlign, an alignment-based bioinformatic tool for comparing chromatin state sequences.

1.1 P-value-free FDR control on high-throughput data from two conditions

The first part of my dissertation focuses on false discovery control on high-throughput data with two conditions. The false discovery rate (FDR) [7] has been developed as a statistical criterion for ensuring discoveries' reliability. The FDR technically is defined as the expected proportion of uninteresting features among the discoveries. FDR control refers to the goal of finding discoveries such that the FDR is under a pre-specified threshold (e.g., 0.05). Existing computational methods for FDR control primarily rely on valid high-resolution p-value calculations. Specifically, p-values are first calculated, one per biological feature (e.g., a gene), and are thresholded using predominantly the Benjamini-Horchberg (BH) procedure [7], the Storey's q-values [8] or other FDR control methods [9–12]. All these methods set a p-value cutoff based on the pre-specified FDR threshold. However, the calculation of p-values requires either distributional assumptions, which are often questionable, or large numbers of replicates, which are often unachievable in biological studies. Due to these limitations of p-value-based methods in high-throughput biological data analysis, bioinformatics tools often which consequently leads to unreliable FDR control. Therefore, p-value-free FDR control is desirable, as it would make high-throughput data analysis more transparent and thus improve the reproducibility of scientific research.

In Chapter 2, we propose Clipper, a model-free and p-value-free FDR control framework for analyzing high-throughput data with two conditions [13]. Clipper is a robust and flexible framework that applies to different analysis tasks and that works for high-throughput data with various characteristics. In comprehensive simulation and real-data benchmarking, Clipper outperforms existing generic FDR control methods and specific bioinformatics tools, including peak calling from ChIP-seq data, and differentially expressed gene identification from bulk and single-cell RNA-seq data. Our results demonstrate Clipper's flexibility and reliability for FDR control and its broad applications in high-throughput data analysis.

1.2 Alignment-based bioinformatic tool for comparing chromatin state sequences

The second part of my dissertation focuses on constructing a measurement of the similarity or difference between two genomic regions based from multi-track epigentic data. A series of computational methods, including ChromHMM [14], and Segway [15], have been developed to build a genome-wide chromatin state annotation, where distinct chromatin states have demonstrated diverse regulatory and transcriptional signals [16–18]. In these methods, each epigenome is segmented into non-overlapping regions, and a single-track chromatin state sequence is constructed by compressing the multi-track epigenetic activities (e.g., DNA methylation and histone modifications) in various ways. With the chromatin state annotations, we can reduce the challenging question of comparing multi-track epigenomic signals into a simpler task of comparing two chromatin state sequences.

Based on existing chromatin state annotations, previous work has studied similarities and differences of human tissue and cell types in terms of epigenomic signals in specific functional genomic elements (e.g., promoters and enhancers), as well as the tissue and cell specificity of these elements, using the Pearson correlation coefficients [19, 20] or a newly developed epigenome overlap measure (EPOM) [21]. However, former epigenome comparative studies failed to effectively incorporate the sequential information of chromatin states, which, however, we believe are highly likely to contain critical information on gene regulatory mechanisms.

Many sequence alignment methods have been developed over the past decades to measure the similarity between DNA/RNA sequences [22, 23]. With the development of these algorithms, sequence alignment tools have become indispensable in almost all modern biological research. Motivated by the enormous successes of sequence alignment algorithms in comparing nucleotide and protein sequences [24], in Chapter 3, we propose a novel computational method, Epigenome Alignment (EpiAlign), to compare two genomic regions by aligning their chromatin state sequences. EpiAlign compares two chromatin state sequences by calculating a local alignment score. It also allows the search of genomic regions (i.e., "hits") whose chromatin state sequences are similar to those of a query region. Aligned chromatin state sequences are expected to have similar biological functions. EpiAlign is flexible in performing the chromatin state sequence alignment either within an epigenome, i.e., a tissue or cell, or between two epigenomes. From the alignment results of EpiAlign, users can identify common chromatin state patterns or differential genomic regions across conditions to investigate the function of genomic regions.

1.3 Summary

During my doctoral study, I have developed the aforementioned two statistical methods that both involve high-throughput data analysis comparing two conditions. The details of these projects will be described in Chapter 2–3 of this dissertation.

CHAPTER 2

P-value free FDR control on high-throughput biological data from two conditions

This is a collaborative work with my labmate Dr. Yiling Chen [13], so it is also a major part of her dissertation [25]. For the methodology development, Yiling and I made equal contributions, so our dissertations are similar in the background part and the methodology part. For the results, I am the major contributor. Specifically, I contributed to all the simulation analysis and three omics data analysis, including peak calling from ChIP-seq data, differentially expressed gene (DEG) identification from real bulk RNA-seq data, and DEG identification from signle-cell RNA-seq data. Yiling contributed to the three other omics data analysis, including peptide identification from mass spectrometry data, differentially expressed gene identification from synthetic bulk RNA-seq data, and differentially interacting chromatin region identification from Hi-C data.

2.1 Introduction

High-throughput technologies are widely used to measure system-wide biological features, such as genes, genomic regions, and proteins ("high-throughput" means the number of features is large, at least in thousands). The most common goal of analyzing high-throughput data is to contrast two conditions so as to reliably screen "interesting features," where "interesting" means "enriched" or "differential." "Enriched features" are defined to have higher expected measurements (without measurement errors) under the experimental (i.e., treatment) condition than the background (i.e., the negative control) condition. The detection of enriched features is called "enrichment analysis." For example, typical enrichment analyses include calling protein-binding sites in a genome from chromatin immunoprecipitation sequencing (ChIP-seq) data [26, 27]. In contrast, “differential features” are defined to have different expected measurements between two conditions, and their detection is called “differential analysis.” For example, popular differential analyses include the identification of differentially expressed genes (DEGs) from genome-wide gene expression data (e.g., microarray and RNA sequencing (RNA-seq) data [28–34]) (Fig. 2.1a). In most scientific research, the interesting features only constitute a small proportion of all features, and the remaining majority is referred to as “uninteresting features.”

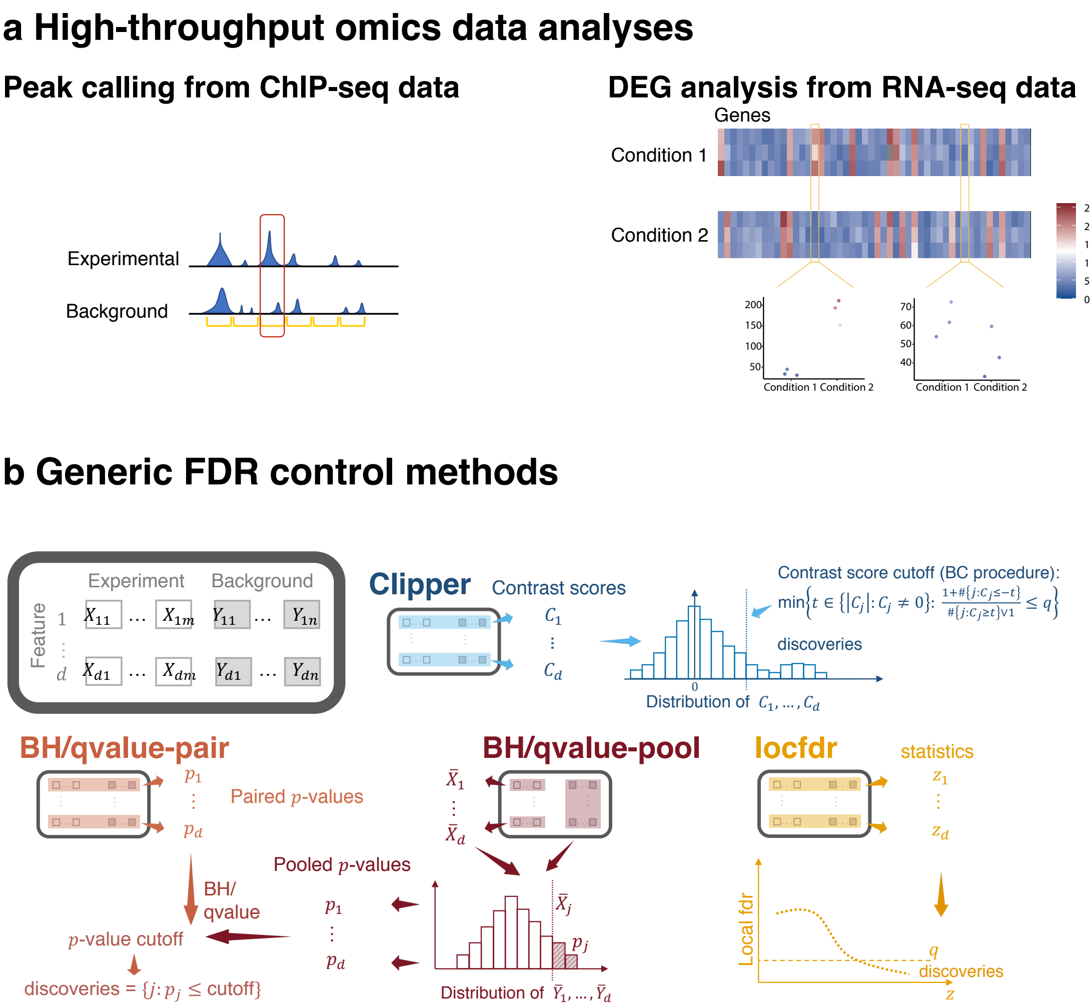

a High-throughput omics data analyses
Peak calling from ChIP-seq data
Experimental
Background
DEG analysis from RNA-seq data
Genes
Condition 1
Condition 2
250
200
150
100
50
0
200
150
100
50
Condition 1 Condition 2
70
60
50
40
Condition 1 Condition 2
b Generic FDR control methods
Experiment
Background
Feature
1
d
X_{11} … X_{1m} Y_{11} … Y_{1n}
X_{d1} … X_{dm} Y_{d1} … Y_{dn}
Clipper
Contrast scores
C_1
C_d
Contrast score cutoff (BC procedure):
$\min\left\{t \in \{|C_j| : C_j \neq 0\} : \frac{1+\#\{j:C_j \le -t\}}{\#\{j:C_j \ge t\} \vee 1} \le q\right\}$
discoveries
0
Distribution of $C_1, \dots, C_d$
BH/qvalue-pair
p_1
p_d
Paired p-values
BH/
qvalue
p-value cutoff
discoveries = $\{j : p_j \le \text{cutoff}\}$
BH/qvalue-pool
$\bar{X}_1$
$\bar{X}_d$
Pooled p-values
p_1
p_d
$\bar{X}_j$
p_j
Distribution of $\bar{Y}_1, \dots, \bar{Y}_d$
locfdr
statistics
z_1
z_d
Local fdr
q
discoveries
z

Figure 2.1: High-throughput omics data analyses and generic FDR control methods.

(a) Illustration of two common high-throughput omics data analyses: peak calling from ChIP-seq data, and DEG analysis from RNA-seq data. In these two analyses, the corresponding features are genomic regions (yellow intervals), and genes (columns in the heatmaps) (b) Illustration of Clipper and five generic FDR control methods: BH-pair (and qvalue-pair), BH-pool (and qvalue-pool), and locfdr.

The identified features, also called the "discoveries" from enrichment or differential analysis, are subject to further investigation and validation. Hence, to reduce experimental validation that is often laborious or expensive, researchers demand reliable discoveries that contain few false discoveries. Accordingly, the false discovery rate (FDR) [7] has been developed as a statistical criterion for ensuring discoveries' reliability. The FDR technically is defined as the expected proportion of uninteresting features among the discoveries under the frequentist statistical paradigm. In parallel, under the Bayesian paradigm, other criteria have been developed, including the Bayesian false discovery rate [35], the local false discovery rate (local fdr) [36], and the local false sign rate [37]. Among all these frequentist and Bayesian criteria, the FDR is the dominant criterion for setting thresholds in biological data analysis [26, 34, 38–44] and is thus the focus of this paper.

FDR control refers to the goal of finding discoveries such that the FDR is under a pre-specified threshold (e.g., 0.05). Existing computational methods for FDR control primarily rely on p-values, one per feature. Among the p-value-based methods, the most classic and popular ones are the Benjamini-Hochberg (BH) procedure [7] and the Storey's q-value [8]; later development introduced methods that incorporate feature weights [9] or covariates (e.g., independent hypothesis weighting (IHW) [10], adaptive p-value thresholding [11], and Boca and Leek's FDR regression [12]) to boost the detection power. All these methods set a p-value cutoff based on the pre-specified FDR threshold. However, the calculation of p-values requires either distributional assumptions, which are often questionable, or large numbers of replicates, which are often unachievable in biological studies (see Results). Due to these limitations of p-value-based methods in high-throughput biological data analysis, bioinformatics tools often output ill-posed p-values. This issue is evidenced by serious concerns about the widespread miscalculation and misuse of p-values in the scientific community [45]. As a result, bioinformatics tools using questionable p-values either cannot reliably control the FDR to a target level [43] or lack power to make discoveries [46]; see Results. Therefore, p-value-free control of FDR is desirable, as it would make data analysis more transparent and thus improve the reproducibility of scientific research.

Although p-value-free FDR control has been implemented in the MACS2 method for

ChIP-seq peak calling [26] and the SAM method for microarray DEG identification [47], these two methods are restricted to specific applications and lack theoretical guarantee for FDR control[1]. More recently, the Barber-Candès (BC) procedure has been proposed to achieve theoretical FDR control without using p-values [50], and it has been shown to perform comparably to the BH procedure with well-calibrated p-values [51]. The BC procedure is advantageous because it does not require well-calibrated p-values, so it holds tremendous potential in various high-throughput data analyses where p-value calibration is challenging [52]. For example, a recent paper has implemented a generalization of the BC procedure to control the FDR in peptide identification from MS data [53].

Inspired by the BC procedure, we propose a general statistical framework Clipper to provide reliable FDR control for high-throughput biological data analysis, without using p-values or relying on specific data distributions. Clipper is a robust and flexible framework that applies to both enrichment and differential analyses and that works for high-throughput data with various characteristics, including data distributions, replicate numbers (from one to multiple), and outlier existence.

2.2 The Clipper methodology

Notations and assumptions

We first introduce notations and assumptions used in Clipper. While the differential analysis treats the two conditions symmetric, the enrichment analysis requires one condition to be the experimental condition (i.e., the condition of interest) and the other condition to be the background condition (i.e., the negative control). For simplicity, we use the same set of notations for both analyses. For two random vectors $\boldsymbol{X} = (X_1, \ldots, X_m)^\top$ and $\boldsymbol{Y} = (Y_1, \ldots, Y_n)^\top$, we write $\boldsymbol{X} \perp \boldsymbol{Y}$ if X_i is independent of Y_j for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$. To avoid confusion, we use $\text{card}(A)$ to denote the cardinality of a set A and $|c|$ to denote the absolute value of a scalar c. We define $a \vee b := \max(a, b)$.

[1] Although later works have studied some theoretical properties of SAM, they are not about the exact control of the FDR [48, 49].

Clipper only requires two inputs: the target FDR threshold $q \in (0, 1)$ and the input data. Regarding the input data, we use d to denote the number of features with measurements under two conditions, and we use m and n to denote the numbers of replicates under the two conditions. For each feature $j = 1, \ldots, d$, we use $\boldsymbol{X}_j = (X_{j1}, \ldots, X_{jm})^\top \in \mathbb{R}^m$ and $\boldsymbol{Y}_j = (Y_{j1}, \ldots, Y_{jn})^\top \in \mathbb{R}^n$ to denote its measurements under the two conditions, where $\mathbb{R}$ denotes the set of non-negative real numbers. We assume that all measurements are non-negative, as in the case of most high-throughput experiments. (If this assumption does not hold, transformations can be applied to make data satisfy this assumption.)

Clipper has the following assumptions on the joint distribution of $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_d, \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_d$. For $j = 1, \ldots, d$, Clipper assumes that $X_{j1}, \ldots, X_{jm}$ are identically distributed, so are $Y_{j1}, \ldots, Y_{jn}$. Let $\mu_{Xj} = \mathbb{E}[X_{j1}]$ and $\mu_{Yj} = \mathbb{E}[Y_{j1}]$ denote the expected measurement of feature j under the two conditions, respectively. Then conditioning on $\{\mu_{Xj}\}_{j=1}^d$ and $\{\mu_{Yj}\}_{j=1}^d$,

$$\begin{gathered} X_{j1}, \cdots, X_{jm}, Y_{j1}, \cdots, Y_{jn} \text{ are mutually independent}\,; \qquad (2.1) \\ \boldsymbol{X}_j \perp \boldsymbol{X}_k, \boldsymbol{Y}_j \perp \boldsymbol{Y}_k \text{ and } \boldsymbol{X}_j \perp \boldsymbol{Y}_k\,,\ \forall j, k = 1, \ldots, d. \end{gathered}$$

An enrichment analysis aims to identify interesting features with $\mu_{Xj} > \mu_{Yj}$ (with $\boldsymbol{X}_j$ and $\boldsymbol{Y}_j$ defined as the measurements under the experimental and background conditions, respectively), while a differential analysis aims to call interesting features with $\mu_{Xj} \neq \mu_{Yj}$. We define $\mathcal{N} := \{j : \mu_{Xj} = \mu_{Yj}\}$ as the set of uninteresting features and denote $N := \text{card}(\mathcal{N})$. In both analyses, Clipper further assumes that an uninteresting feature j satisfies

$$X_{j1}, \cdots, X_{jm}, Y_{j1}, \cdots, Y_{jn} \text{ are identically distributed}\,, \forall j \in \mathcal{N}\,. \qquad (2.2)$$

Clipper consists of two main steps: construction and thresholding of contrast scores. First, Clipper computes contrast scores, one per feature, as summary statistics that reflect the extent to which features are interesting. Second, Clipper establishes a contrast-score cutoff and calls as discoveries the features whose contrast scores exceed the cutoff.

To construct contrast scores, Clipper uses two summary statistics $t(\cdot, \cdot) : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$

to extract data information regarding whether a feature is interesting or not:

$$t^{\text{minus}}(\boldsymbol{x}, \boldsymbol{y}) := \bar{x} - \bar{y}\,; \tag{2.3}$$

$$t^{\text{max}}(\boldsymbol{x}, \boldsymbol{y}) := \max\left(\bar{x}, \bar{y}\right) \cdot \text{sign}\left(\bar{x} - \bar{y}\right)\,, \tag{2.4}$$

where $\boldsymbol{x} = (x_1, \ldots, x_m)^\top \in \mathbb{R}^m$, $\boldsymbol{y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$, $\bar{x} = \sum_{i=1}^m x_i/m$, $\bar{y} = \sum_{i=1}^n y_i/n$, and $\text{sign}(\cdot) : \mathbb{R} \to \{-1, 0, 1\}$ with $\text{sign}(x) = 1$ if $x > 0$, $\text{sign}(x) = -1$ if $x < 0$, and $\text{sign}(x) = 0$ otherwise.

Notably, other summary statistics can also be used to construct contrast scores. For example, an alternative summary statistic is the t statistic from the two-sample t test:

$$t^{\text{t}}(\boldsymbol{x}, \boldsymbol{y}) := \frac{\bar{x} - \bar{y}}{\sqrt{\frac{\sum_{i=1}^m (x_i - \bar{x})^2 + \sum_{i=1}^n (y_i - \bar{y})^2}{m+n-2}}}\,. \tag{2.5}$$

Then we introduce how Clipper works in three analysis tasks: the enrichment analysis with equal numbers of replicates under two conditions ($m = n$), the enrichment analysis with different numbers of replicates under two conditions ($m \neq n$), and the differential analysis (when $m + n > 2$).

Enrichment analysis with equal numbers of replicates ($m = n$)

Under the enrichment analysis, we assume that $\boldsymbol{X}_j \in \mathbb{R}^m$ and $\boldsymbol{Y}_j \in \mathbb{R}^n$ are the measurements of feature j, $j = 1, \ldots, d$, under the experimental and background conditions with m and n replicates, respectively. We start with the simple case when $m = n$. Clipper defines a contrast score C_j of feature j in one of two ways:

$$C_j := t^{\text{minus}}(\boldsymbol{X}_j, \boldsymbol{Y}_j) \qquad \textbf{minus contrast score}\,, \tag{2.6}$$

or

$$C_j := t^{\text{max}}(\boldsymbol{X}_j, \boldsymbol{Y}_j) \qquad \textbf{maximum contrast score}\,. \tag{2.7}$$

Fig. 2.2a shows a cartoon illustration of contrast scores when $m = n = 1$. Accordingly, a large positive value of C_j bears evidence that $\mu_{Xj} > \mu_{Yj}$. Motivated by Barber and Candès [50], Clipper uses the following procedure to control the FDR under the target level $q \in (0, 1)$.

Definition 1 (Barber-Candès (BC) procedure for thresholding contrast scores [50])**.** *Given contrast scores $\{C_j\}_{j=1}^d$, $\mathcal{C} = \{|C_j| : C_j \neq 0\ ;\ j = 1, \ldots, d\}$ is defined as the set of non-zero absolute values of C_j's. The BC procedure finds a contrast-score cutoff T^{BC} based on the target FDR threshold $q \in (0, 1)$ as*

$$T^{BC} := \min \left\{ t \in \mathcal{C} : \frac{\operatorname{card}(\{j : C_j \leqslant -t\}) + 1}{\operatorname{card}(\{j : C_j \geqslant t\}) \vee 1} \leqslant q \right\} \tag{2.8}$$

and outputs $\left\{j : C_j \geqslant T^{BC}\right\}$ as discoveries.

Enrichment analysis with any numbers of replicates m and n

When $m \neq n$, Clipper constructs contrast scores via permutation of replicates across conditions. The idea is that, after permutation, every feature becomes uninteresting and can serve as its own negative control.

Definition 2 (Permutation)**.** *We define σ as permutation, i.e., a bijection from the set $\{1, \cdots, m+n\}$ onto itself, and we rewrite the data $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_d, \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_d$ into a matrix $\mathbf{W}$:*

$$\mathbf{W} = \begin{bmatrix} W_{11} & \cdots & W_{1m} & W_{1(m+1)} & \cdots & W_{1(m+n)} \\ & \vdots & & & \vdots & \\ W_{d1} & \cdots & W_{dm} & W_{d(m+1)} & \cdots & W_{d(m+n)} \end{bmatrix} := \begin{bmatrix} X_{11} & \cdots & X_{1m} & Y_{11} & \cdots & Y_{1n} \\ & \vdots & & & \vdots & \\ X_{d1} & \cdots & X_{dm} & Y_{d1} & \cdots & Y_{dn} \end{bmatrix}.$$

We then apply σ to permute the columns of $\mathbf{W}$ and obtain

$$\mathbf{W}_\sigma := \begin{bmatrix} W_{1\sigma(1)} & \cdots & W_{1\sigma(m)} & W_{1\sigma(m+1)} & \cdots & W_{1\sigma(m+n)} \\ & \vdots & & & \vdots & \\ W_{d\sigma(1)} & \cdots & W_{d\sigma(m)} & W_{d\sigma(m+1)} & \cdots & W_{d\sigma(m+n)} \end{bmatrix},$$

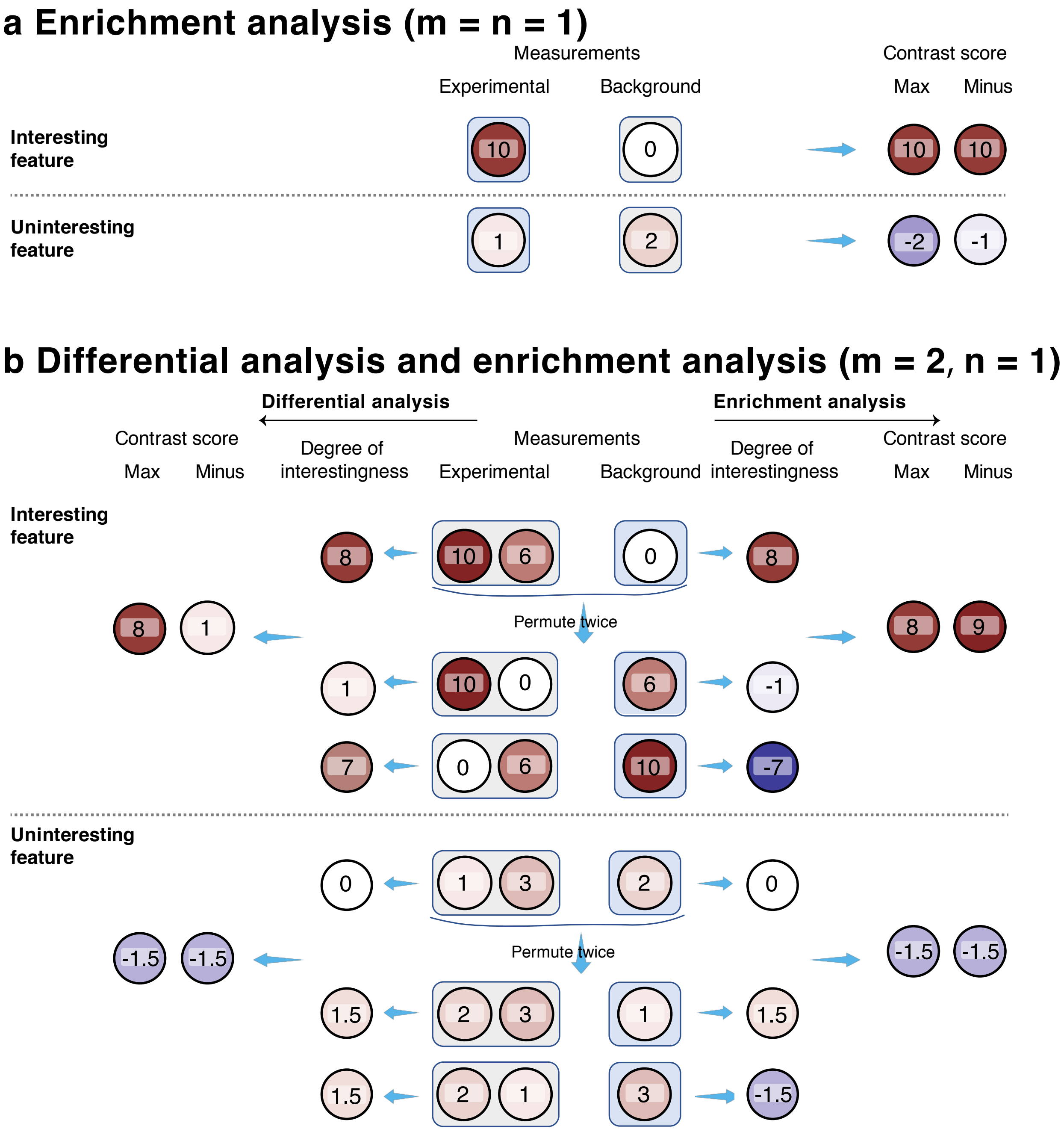


Figure 2.2: Illustration of the construction of contrast scores.

(a) 1vs1 enrichment analysis; (b) 2vs1 differential analysis (left) or enrichment analysis (right). In each panel, an interesting feature (top) and an uninteresting feature (bottom) are plotted for contrast; both features have measurements under the experimental and background conditions. In (a), each feature's measurements are summarized into a maximum (max) contrast score or a minus contrast score. In (b), each feature's measurements are permuted across the two conditions, resulting in two sets of permuted measurements. Then for each feature, we calculate its degrees of interestingness (as the difference that equals the average of experimental measurements minus the average of background measurements (in enrichment analysis; right), or the absolute value of the difference (in differential analysis; left)) from its original measurements and permuted measurements, respectively. Finally, we summarize each feature's degrees of interestingness into a maximum (max) contrast score or a minus contrast score.

from which we obtain the permuted measurements $\left\{(\boldsymbol{X}_j^\sigma, \boldsymbol{Y}_j^\sigma)\right\}_{j=1}^d$*, where*

$$\begin{aligned}\boldsymbol{X}_j^\sigma &:= \left(W_{j\sigma(1)}, \ldots, W_{j\sigma(m)}\right)^\top , \\ \boldsymbol{Y}_j^\sigma &:= \left(W_{j\sigma(m+1)}, \ldots, W_{j\sigma(m+n)}\right)^\top . \end{aligned} \tag{2.9}$$

In the enrichment analysis, if two permutations σ and σ' satisfy that

$$\{\sigma(1), \cdots, \sigma(m)\} = \{\sigma'(1), \cdots, \sigma'(m)\} ,$$

then we define σ and σ' to be in one equivalence class. That is, permutations in the same equivalence class lead to the same division of $m+n$ replicates (from the two conditions) into two groups with sizes m and n. In total, there are $\binom{m+n}{m}$ equivalence classes of permutations.

We define σ_0 as the identity permutation such that $\sigma_0(i) = i$ for all $i \in \{1, \cdots, m+n\}$. In addition, Clipper randomly samples h equivalence classes $\sigma_1, \ldots, \sigma_h$ with equal probabilities without replacement from the other $h_{\max} := \binom{m+n}{m} - 1$ equivalence classes (after excluding the equivalence class containing σ_0). Note that $h_{\max}$ is the maximum value h can take.

Clipper then obtains $\left\{(\boldsymbol{X}_j^{\sigma_0}, \boldsymbol{Y}_j^{\sigma_0}), (\boldsymbol{X}_j^{\sigma_1}, \boldsymbol{Y}_j^{\sigma_1}), \cdots, (\boldsymbol{X}_j^{\sigma_h}, \boldsymbol{Y}_j^{\sigma_h})\right\}_{j=1}^d$, where $(\boldsymbol{X}_j^{\sigma_\ell}, \boldsymbol{Y}_j^{\sigma_\ell})$ are the permuted measurements based on σ_ℓ, $\ell = 0, 1, \ldots, h$. Then Clipper computes $T_j^{\sigma_\ell} := t^{\text{minus}}(\boldsymbol{X}_j^{\sigma_\ell}, \boldsymbol{Y}_j^{\sigma_\ell})$ to indicate the degree of "interestingness" of feature j reflected by $(\boldsymbol{X}_j^{\sigma_\ell}, \boldsymbol{Y}_j^{\sigma_\ell})$. Note that Clipper chooses t^{minus} instead of $t^{\max}$ because empirical evidence shows that t^{minus} leads to better power. Sorting $\{T_j^{\sigma_\ell}\}_{\ell=0}^h$ gives

$$T_j^{(0)} \geqslant T_j^{(1)} \geqslant \cdots \geqslant T_j^{(h)} .$$

Then Clipper defines the contrast score of feature j, $j = 1, \ldots, d$, in one of two ways:

$$C_j := \begin{cases} T_j^{(0)} - T_j^{(1)} & \text{if } T_j^{(0)} = T_j^{\sigma_0} \\ T_j^{(1)} - T_j^{(0)} & \text{otherwise} \end{cases} \qquad \textbf{minus contrast score}\,, \tag{2.10}$$

or

$$C_j := \begin{cases} \left|T_j^{(0)}\right| & \text{if } T_j^{(0)} = T_j^{\sigma_0} > T_j^{(1)} \\ 0 & \text{if } T_j^{(0)} = T_j^{(1)} \\ -\left|T_j^{(0)}\right| & \text{otherwise} \end{cases} \qquad \textbf{maximum contrast score}\,. \tag{2.11}$$

The intuition behind the contrast scores is that, if $C_j < 0$, then $T_j^{(0)} \neq T_j^{\sigma_0}$, which means that at least one of $T_j^{\sigma_1}, \ldots, T_j^{\sigma_h}$ (after random permutation) is greater than $T_j^{\sigma_0}$ calculated from the original data (identity permutation), suggesting that feature j is likely an uninteresting feature in enrichment analysis. Fig. 2.2b (right) shows a cartoon illustration of contrast scores when $m = 2$ and $n = 1$. Motivated by Gimenez and Zou [54], we propose the following procedure for Clipper to control the FDR under the target level $q \in (0, 1)$.

Definition 3 (Gimenez-Zou (GZ) procedure for thresholding contrast scores [54])**.** *Given* $h \in \{1, \ \cdots, h_{\max}\}$ *and contrast scores* $\{C_j\}_{j=1}^d$, $\mathcal{C} = \{|C_j| : C_j \neq 0\ ;\ j = 1, \ldots, d\}$ *is defined as the set of non-zero absolute values of* C_j*'s. The GZ procedure finds a contrast-score cutoff* T^{GZ} *based on the target FDR threshold* $q \in (0, 1)$ *as:*

$$T^{GZ} := \min\left\{ t \in \mathcal{C} : \frac{\frac{1}{h} + \frac{1}{h}\text{card}\left(\{j : C_j \leqslant -t\}\right)}{\text{card}\left(\{j : C_j \geqslant t\}\right) \vee 1} \leqslant q \right\} \tag{2.12}$$

and outputs $\left\{j : C_j \geqslant T^{GZ}\right\}$ *as discoveries.*

Differential analysis with $m + n > 2$

For differential analysis, Clipper also uses permutation to construct contrast scores. When $m \neq n$, the equivalence classes of permutations are defined the same as for the enrichment analysis with $m \neq n$. When $m = n$, there is a slight change in the definition of equivalence classes of permutations: if σ and σ' satisfy that

$$\{\sigma(1), \cdots, \sigma(m)\} = \{\sigma'(1), \cdots, \sigma'(m)\} \text{ or } \{\sigma'(m+1), \cdots, \sigma'(2m)\},$$

then we say that σ and σ' are in one equivalence class. In total, there are $h_{\text{total}} := \binom{m+n}{m}$ (when $m \neq n$) or $\binom{2m}{m}/2$ (when $m = n$) equivalence classes of permutations. Hence, to have more than one equivalence class, we cannot perform differential analysis with $m = n = 1$; in other words, the total number of replicates $m + n$ must be at least 3.

Then Clipper randomly samples $\sigma_1, \ldots, \sigma_h$ with equal probabilities without replacement from the $h_{\max} := h_{\text{total}} - 1$ equivalence classes that exclude the class containing σ_0, i.e., the identity permutation. Note that $h_{\max}$ is the maximum value h can take. Next, Clipper computes $T_j^{\sigma_\ell} := \left|t^{\text{minus}}(\boldsymbol{X}_j^{\sigma_\ell}, \boldsymbol{Y}_j^{\sigma_\ell})\right|$, where $\boldsymbol{X}_j^{\sigma_\ell}$ and $\boldsymbol{Y}_j^{\sigma_\ell}$ are the permuted data defined in (2.9), and it defines C_j as the contrast score of feature j, $j = 1, \ldots, d$, in the same ways as in (2.10) or (2.11). Fig. 2.2b (left) shows a cartoon illustration of contrast scores when $m = 2$ and $n = 1$.

Same as in the enrichment analysis with $m \neq n$, Clipper also uses the GZ procedure [54] to set a cutoff on contrast scores to control the FDR under the target level $q \in (0, 1)$.

Granted, when we use permutations to construct contrast scores in the GZ procedure, we can convert contrast scores into permutation-based p-values (see Supp. S2.5.1.1). However, when the numbers of replicates are small, the number of possible permutations is small, so permutation-based p-values would have a low resolution (e.g., when $m = 2$ and $n = 1$, the number of non-identity permutations is only 2). Hence, applying the BH procedure to the permutation-based p-values would result in almost no power. Although Yekutieli and Benjamini proposed another thresholding procedure for permutation-based p-values [55], it still requires the number of permutations to be large to obtain a reliable FDR control. Furthermore, if we apply the SeqStep+ procedure by Barber and Candés [50] to permutation-based p-values, it would be equivalent to our application of the GZ procedure to contrast scores (Supp. Section S2.5.1.1).

For both differential and enrichment analyses, the two contrast scores (minus and maximum) can both control the FDR. Based on the power comparison results in Supp. Section S2.5.2 and Supp. Figs. 2.28–2.31, Clipper has the following default choice of contrast score: for the enrichment analysis when two conditions have the same number of replicates ("Enrichment analysis with equal numbers of replicates ($m = n$)" in Methods), Clipper uses the BC procedure with the minus contrast score; for the enrichment analysis when two conditions have different numbers of replicates ("Enrichment analysis with any numbers of replicates m and n" in Methods) or the differential analysis ("Differential analysis with $m + n > 2$" in Methods), Clipper uses the GZ procedure with maximum contrast score.

2.2.1 Clipper variant algorithms

For nomenclature, we assign the following names to Clipper variant algorithms, each of which combines a contrast score definition with a thresholding procedure.

- **Clipper-diff-BC**: difference contrast score $C_j = t^{\text{diff}}(\boldsymbol{X}_j, \boldsymbol{Y}_j)$ (2.6) and BC procedure (Definition 1);
- **Clipper-diff-GZ**: difference contrast score $\tau_j = T_j^{(0)} - T_j^{(1)}$ (2.10) and GZ procedure (Definition 3);
- **Clipper-max-BC**: maximum contrast score $C_j = t^{\max}(\boldsymbol{X}_j, \boldsymbol{Y}_j)$ (2.7) and BC procedure;
- **Clipper-max-GZ**: maximum contrast score $\tau_j = T_j^{(0)}$ (2.11) and GZ procedure.

2.2.2 `R` package "Clipper"

In the `R` package `Clipper`, the default implementation is as follows. Based on the power comparison results in our manuscripts Ge et al. [13], Clipper uses Clipper-diff-BC as the default algorithm for the enrichment analysis with equal numbers of replicates; when there are no discoveries, Clipper suggests users to increase the target FDR threshold q or to use the Clipper-diff-aBH algorithm with the current q. For the enrichment analysis with different numbers of replicates under two conditions or the differential analysis, Clipper uses the Clipper-max-GZ algorithm by default.

2.3 Application of Clipper on simulation and omics data analysis

To verify Clipper's performance, we designed comprehensive simulation studies to benchmark Clipper against existing generic FDR control methods (Supp. Section S2.5.1). We also benchmarked Clipper against bioinformatics tools in studies including peak calling from ChIP-seq data, and DEG identification from bulk or single-cell RNA-seq data.

Clipper has verified FDR control and power advantage in simulation

Simulation is essential because we can generate numerous datasets from the same distribution with known truths to calculate the FDR, which is not observable from real data. Our simulation covers both enrichment and differential analyses. In enrichment analysis, we consider four "experimental designs": 1vs1 design (one replicate per condition), 2vs1 design (two and one replicates under the experimental and background conditions, respectively), 3vs3 design (three replicates per condition), and 10vs10 design (ten replicates per condition). In differential analysis, since Clipper requires that at least one condition has two replicates, we only consider the 2vs1 and 3vs3 designs. For each analysis and design, we simulated data from three "distributional families"—Gaussian, Poisson, and negative binomial—for individual features under two "background scenarios" (i.e., scenarios of the background condition): homogeneous and heterogeneous. Under the homogeneous scenario, all features' measurements follow the same distribution under the background condition; otherwise, we are under the heterogeneous scenario, which is ubiquitous in applications, e.g., identifying DEGs from RNA-seq data and calling protein-binding sites from ChIP-seq data. By simulation setting, we refer to a combination of an experimental design, a distributional family, and a background scenario. The details of simulation settings are described in Supp. Section S2.5.3.

For both enrichment and differential analyses and each simulation setting, we compared

Clipper against generic FDR control methods, including p-value-based methods and local-fdr-based methods. The p-value-based methods include BH-pair, BH-pool, qvalue-pair, and qvalue-pool, where "BH" and "qvalue" stand for p-value thresholding procedures, and "pair" and "pool" represent the paired and pooled p-value calculation approaches, respectively. The local-fdr-based methods include locfdr-emp and locfdr-swap, where "emp" and "swap" represent the empirical null and swapping null local-fdr calculation approaches, respectively. See Online Methods for detail.

The comparison results are in Fig. 2.3 and Supp. Figs. 2.7–2.17. A good FDR control method should have actual FDR no larger than the target FDR threshold and achieve high power. The results show that Clipper controls the FDR and is overall more powerful than other methods, excluding those that fail to control the FDR, under all settings. Clipper is also shown to be more robust to the number of features and the existence of outliers than other methods. In detail, in both enrichment analyses (1vs1, 2vs1, 3vs3, and 10vs10 designs) and differential analyses (2vs1 and 3vs3 designs), Clipper consistently controls the FDR, and it is more powerful than the generic methods in most cases under the realistic, heterogeneous background, where features do not follow the same distribution under the background condition. Under the idealistic, homogeneous background, Clipper is still powerful and only second to BH-pool and qvalue-pool, which, however, cannot control the FDR under the heterogeneous background.

Here we summarize the performance of the generic FDR control methods. First, the two p-value-based methods using the pooled approach, BH-pool and qvalue-pool, are the most powerful under the idealistic, homogeneous background, which is their inherent assumption; however, they cannot control the FDR under the heterogeneous background (Fig. 2.3b). Besides, they cannot control the FDR when the number of features is small (Fig. 2.3a and Supp. Fig. 2.7). These results show that the validity of BH-pool and qvalue-pool requires a large number of features and the homogeneous background assumption, two requirements that rarely hold in biological applications.

Second, the four p-value-based methods using the paired approach with misspecified models or misformulated tests (BH-pair-mis, qvalue-pair-mis, BH-pair-2as1, and qvalue-pair-

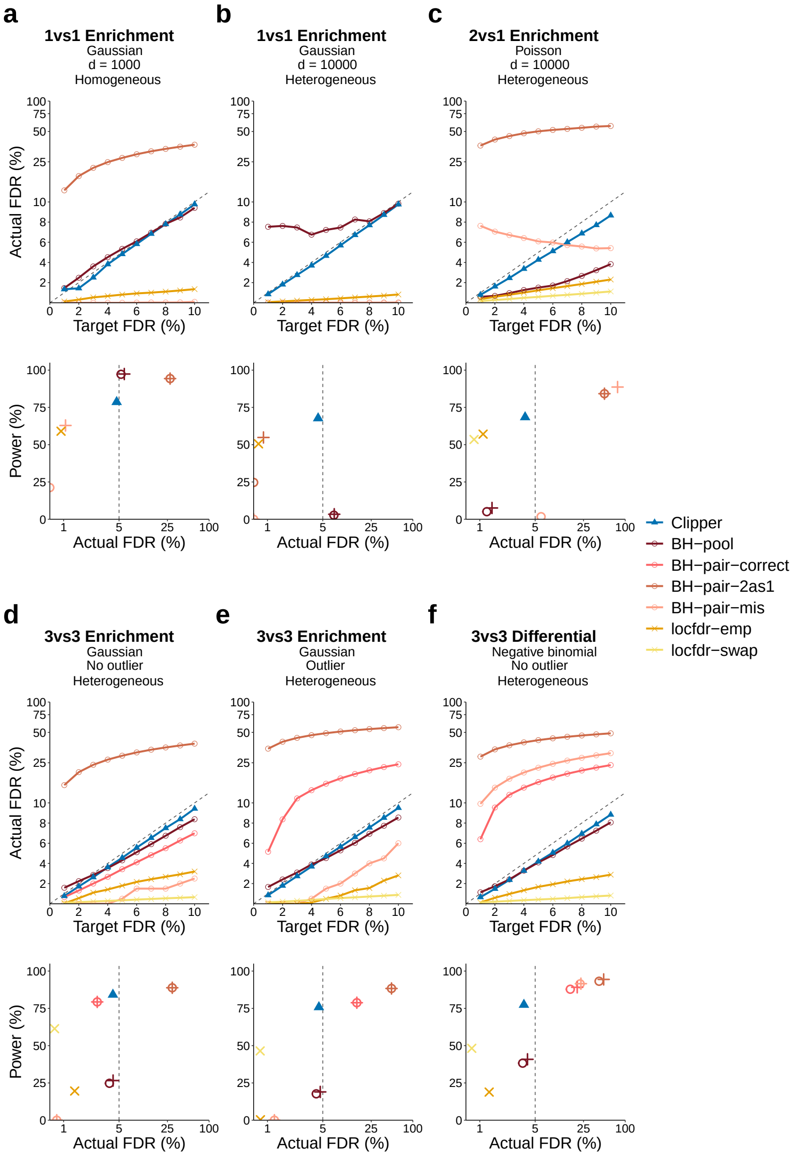


Figure 2.3: Comparison of Clipper with generic FDR control methods in terms of their FDR control and power in six example simulation studies.

(a) 1vs1 enrichment analysis with 1000 features generated from the Gaussian distribution with a homogeneous background; (b) 1vs1 enrichment analysis with 10,000 features generated from the Gaussian distribution with a heterogeneous background; (c) 2vs1 enrichment analysis with 10,000 features generated from the Poisson distribution with a heterogeneous background; (d) 3vs3 enrichment analysis with 10,000 features generated from the Gaussian distribution without outliers and with a heterogeneous background; (e) 3vs3 enrichment analysis with 10,000 features generated from the Gaussian distribution without outliers and with a heterogeneous background; (f) 3vs3 differential analysis with 10,000 features generated from the negative binomial distribution with a heterogeneous background. At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are approximated by the averages of false discovery proportions and power evaluated on 200 simulated datasets. In each panel, the top row shows each method's actual FDRs at target FDR thresholds: whenever the actual FDR is larger than the target FDR (the solid line is higher than the dashed line), FDR control is failed; the bottom row shows each method's actual FDRs and power at the target FDR threshold $q = 5\%$: whenever the actual FDR is greater than q (on the right of the vertical dashed line), FDR control is failed. Under the FDR control, the larger the power, the better. Note that BH-pair-correct is not included in (a)–(c) because it is impossible to correctly specify the model with only one replicate per condition; locfdr-swap is not included in (a)–(b) because it is inapplicable to the 1vs1 design.

2as1; see Online Methods) fail to control the FDR by a large margin in most cases, and rarely when they control the FDR, they lack power (Fig. 2.3c–d and Supp. Figs. 2.7–2.14). These results confirm that the BH-pair and qvalue-pair rely on the correct model specification to control the FDR; however, the correct model specification is hardly achievable with no more than three replicates per condition.

Third, even when models are correctly specified (an idealistic scenario), the p-value-based methods that use the paired approach—BH-pair-correct and qvalue-pair-correct (see Online Methods)—fail to control the FDR in the existence of outliers (Fig. 2.3e and Supp. Figs. 2.9 and 2.13) or for the negative binomial distribution with unknown dispersion (Fig. 2.3f and Supp. Fig. 2.15). It is worth noting that even when they control the FDR, they are less powerful than Clipper in most cases except for the 3vs3 differential analysis with the Poisson distribution (Fig. 2.3d and Supp. Figs. 2.10 and 2.14).

Fourth, the two local-fdr-based methods—locfdr-emp and locfdr-swap—achieve the FDR control under all designs and analyses; however, they are less powerful than Clipper in most cases (Supp. Figs. 2.7–2.10).

Fifth, when the numbers of replicates are large (10vs10 design), non-parametric tests become applicable. We compared Clipper with three BH-pair methods that use different statistical tests: BH-pair-Wilcoxon (the non-parametric Wilcoxon rank-sum test), BH-pair-permutation (the non-parametric permutation test), and BH-pair-parametric (the parametric test based on the correct model specification, equivalent to BH-pair-correct). Although all the three methods control the FDR, they are less powerful than Clipper (Supp. Fig. 2.16).

Moreover, the above five phenomena are consistently observed across the three distributions (Gaussian, Poission, and negative binomial) that we have examined, further confirming the robustness of Clipper.

In addition, for the 3vs3 enrichment analysis, we also varied the proportion of interesting features as 10%, 20%, and 40%. The comparison results in Supp. Fig. 2.9 (columns 1 and 3 for 10%) and Supp. Fig. 2.18 (for 20% and 40%) show that the performance of Clipper is robust to the proportion of interesting features.

The above results are all based on simulations with independent features. To examine the robustness of Clipper, we introduced feature correlations to our simulated data, on which we compared Clipper with other generic FDR control methods. The comparison results in Supp. Fig. 2.17 show that even when the feature independence assumption is violated, Clipper still demonstrates strong performance in both FDR control and power.

Clipper has broad applications in omics data analyses

We then demonstrate the use of Clipper in three omics data applications: peak calling from ChIP-seq data, DEG identification from bulk and single-cell RNA-seq data. The first applications is enrichment analyses, and the last two are differential analyses. In each application, we compared Clipper with mainstream bioinformatics methods to demonstrate Clipper's superiority in FDR control and detection power.

Peak calling from ChIP-seq data (enrichment analysis I)

ChIP-seq is a genome-wide experimental assay for measuring binding intensities of a DNA-associated protein [4], often a transcription factor that activates or represses gene expression [5, 6]. ChIP-seq data are crucial for studying gene expression regulation, and the indispensable analysis is to identify genomic regions with enriched sequence reads in ChIP-seq data. These regions are likely to be bound by the target protein and thus of biological interest. The identification of these regions is termed "peak calling" in ChIP-seq data analysis.

As the identified peaks are subject to experimental validation that is often expensive [56], it is essential to control the FDR of peak identification to reduce unnecessary costs. The two most highly-cited peak-calling methods are MACS2 [26] and [27], both of which claim to control the FDR for their identified peaks. Specifically, both MACS2 and HOMER assume that the read counts for each putative peak (one count per sample/replicate) follow the Poisson distribution, and they use modified paired approaches to assign each putative peak a p-value and a corresponding Storey's q-value. Then given a target FDR threshold $0 < q < 1$, they call the putative peaks with q-values $\leqslant q$ as identified peaks. Despite being popular, MACS2 and HOMER have not been verified for their FDR control, to our knowledge.

To verify the FDR control of MACS2 and HOMER (Supp. Section S2.5.4), we used ENCODE ChIP-seq data of cell line GM12878 [57] and ChiPulate [58], a ChIP-seq data simulator, to generate semi-synthetic data with spiked-in peaks (Supp. Section S2.5.5). We examined the actual FDR and power of MACS2 and HOMER in a range of target FDR thresholds: $q = 1\%, 2\%, \ldots, 10\%$. Fig. 2.6a shows that MACS2 and HOMER cannot control the FDR as standalone peak-calling methods. However, with Clipper as an add-on (Supp. Section S2.5.6), both MACS2 and HOMER can guarantee the FDR control. This result demonstrates the flexibility and usability of Clipper for reducing false discoveries in peak calling analysis.

Technically, the failed FDR control by MACS2 and HOMER is attributable to the likely model misspecification and test misformulation in their use of the paired approach. Both MACS2 and HOMER assume the Poisson distribution for read counts in a putative peak; however, it has been widely acknowledged that read counts are over-dispersed and thus better modeled by the negative binomial distribution [59]. Besides, MACS2 uses one-sample tests to compute p-values when two-sample tests should have been performed. As a result, the p-values of MACS2 and HOMER are questionable, so using their p-values for FDR control would not lead to success. (Note that MACS2 does not use p-values to control the FDR but instead swaps experimental and background samples to calculate the empirical FDR; yet, we emphasize that controlling the empirical FDR does not guarantee the FDR control.) As a remedy, Clipper strengthens both methods to control the FDR while maintaining high power.

It is known that uninteresting regions tend to have larger read counts in the control sample than in the experimental (ChIP) sample, making them more likely to have negative contrast scores than positive ones. However, this phenmenon does not violate Clipper's theoretical assumption (Lemma 1(a) in Supp. Section 2.2), which can be relaxed as we note in Methods.

DEG identification from bulk RNA-seq data (differential analysis I)

RNA-seq data measure genome-wide gene expression levels in biological samples. An important use of RNA-seq data is the DEG analysis, which aims to discover genes whose expression levels change between two conditions. The FDR is a widely used criterion in DEG analysis [28–33].

We compared Clipper with two popular DEG identification methods: edgeR [28] and DESeq2 [29] (Supp. Section S2.5.4). Specifically, when we implemented Clipper, we first performed the trimmed mean of M values (TMM) normalization [60] to correct for batch effects; then we treated genes as features and their normalized expression levels as measurements under two conditions (Supp. Section S2.5.6). We also implemented two versions of DESeq2 and edgeR: with or without IHW, a popular procedure for boosting the power of p-value-based FDR control methods by incorporating feature covariates [10]. In our implementation of the two versions of DESeq2 and edgeR, we used their standard pipelines, including normalization, model fitting, and gene filtering (edgeR only). To verify the FDR control, we generated four realistic synthetic datasets from two real RNA-seq datasets—one from classical and non-classical human monocytes [61] and the other from yeasts with or without *snf2* knockout [62]—using simulation strategies 1 and 2 (Supp. Section S2.5.5).

In detail, in simulation strategy 1, we used bulk RNA-seq samples from two conditions to compute a fold change for every gene between the two conditions; then we defined true DEGs as the genes whose fold changes exceeded a threshold; next, we randomly drew three RNA-seq samples and treated them as replicates from each condition ($m = n = 3$ as in Methods); using those subsampled replicates of two conditions, we preserved the true DEGs' read counts and permuted the read counts of the true non-DEGs, i.e., the genes other than true DEGs, between conditions. In summary, simulation strategy 1 guarantees that the measurements of true non-DEGs are i.i.d., an assumption that Clipper relies on for theoretical FDR control.

In simulation strategy 2, borrowed from a benchmark study [63], we first randomly selected at most 30% genes as true DEGs; next, we randomly drew six RNA-seq samples from one condition (classical human monocytes and yeasts without knockout) and split the samples into two “synthetic conditions,” each with three replicates ($m = n = 3$ as in Methods); then for each true DEG, we multiplied its read counts under one of the two synthetic conditions (randomly picked independently for each gene) by a randomly generated fold change (see Supp. Section S2.5.5); finally, for the true non-DEGs, we preserved their read counts in the six samples. In summary, simulation strategy 2 preserves batch effects, if existent in real data, for the true non-DEGs (the majority of genes). As a result, the semi-synthetic data generated under strategy 2 may violate the Clipper assumption for theoretical FDR control and thus can help evaluate the robustness of Clipper on real data.

The four semi-synthetic datasets have ground truths (true DEGs and non-DEGs) to evaluate each DEG identification method's FDR and power for a range of target FDR thresholds: $q = 1\%, 2\%, \ldots, 10\%$. Our results in Fig. 2.4a and Supp. Figs. 2.21a–2.23a show that Clipper consistently controls the FDR and achieves high power on all four semi-synthetic datasets. In contrast, DESeq2 and edgeR cannot consistently control the FDR except for the yeast semi-synthetic dataset generated under simulation strategy 2. Given the fact that DESeq2 and edgeR do not consistently perform well on the three other semi-synthetic datasets, we hypothesize that their parametric distributional assumptions, if violated on real data, hinder valid FDR control, in line with our motivation for developing Clipper. Furthermore, we observe that adding IHW to edgeR and DESeq2 has negligible effects on the four semi-synthetic datasets.

To further explain why DESeq2 fails to control the FDR, we examined the p-value distributions of 16 non-DEGs that were most frequently identified (from the 100 semi-synthetic datasets generated from the human monocyte dataset using simulation strategy 1) by DESeq2 at the target FDR threshold $q = 0.05$. Our results in Supp. Fig. 2.24 show that the 16 non-DEGs' p-values are non-uniformly distributed with a mode close to 0. Such unusual enrichment of overly small p-values makes these non-DEGs mistakenly called discoveries by DESeq2.

In addition, we compared the DEG ranking by Clipper, edgeR, and DESeq2 in two ways. First, for true DEGs, we compared their ranking by each method with their true ranking based on true expression fold changes (from large to small, as in semi-synthetic

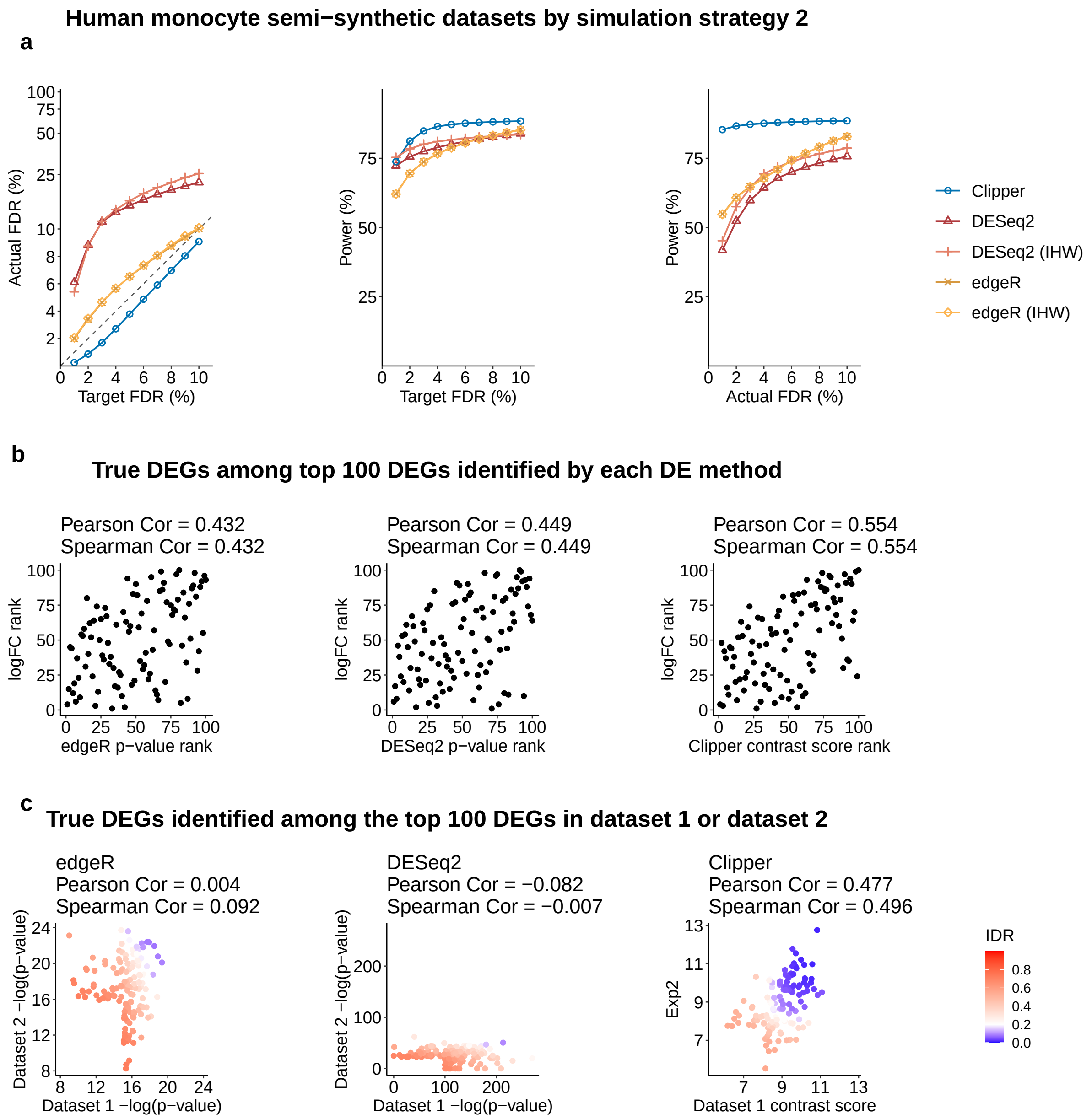


Figure 2.4: Comparison of Clipper and two popular DEG identification methods—edgeR and DESeq2—in DEG analysis on semi-synthetic bulk RNA-seq data (generated from human monocyte real data using simulation strategy 2 in Supp. Section S6.3).

(a) FDR control, power given the same target FDR, and power given the same actual FDR. **(b)** Ranking consistency of the true DEGs among the top 100 DEGs identified by each method. The consistency is defined between the genes' ranking based on edgeR/DESeq2's p-values or Clipper's contrast scores and their ranking based on true expression fold changes. **(c)** Reproducibility between two semi-synthetic datasets as technical replicates. Three reproducibility criteria are used: the IDR, Pearson correlation, and Spearman correalation. Each criterion is calculated for edgeR/DESeq2's p-values or Clipper's contrast scores on the two semi-synthetic datasets. Among the three methods, only Clipper controls the FDR, and Clipper achieves the highest power, the best gene ranking consistency, and the best reproducibility.

data generation in Supp. Section S2.5.5). Specifically, we ranked true DEGs using Clipper's contrast scores (from large to small), edgeR's p-values (from small to large), or DESeq2's p-values (from small to large). Our results in Fig. 2.4b and Supp. Figs. 2.21b–2.23b show that Clipper's contrast scores exhibit the most consistent ranking with the ranking based on true fold changes. Second, to compare the power of Clipper, edgeR, and DESeq2 based on their DEG rankings instead of nominal p-values, we calculated their power under the actual FDRs, which only depend on gene rankings (for the definition of actual FDR, see Supp. Section S2.5.5). Fig. 2.4a and Supp. Figs. 2.21a–2.23a show that, when Clipper, edgeR, and DESeq2 have the same actual FDR, Clipper consistently outperforms edgeR and DESeq2 in terms of power, i.e., Clipper has the most true DEGs in its top ranked genes.

We also compared the reproducibility of Clipper, edgeR, and DESeq2 in the presence of sampling randomness. Specifically, we used two semi-synthetic datasets (generated independently from the same procedure in Supp. Section S2.5.5) as technical replicates and computed Clipper's contrast scores and edgeR's and DESeq's p-values on each dataset. For each method, we evaluated its reproducibility between the two semi-synthetic datasets by computing three criteria—the irreproducibility discovery rate (IDR) [64], Pearson correlation, and Spearman correlation—using its contrast scores or negative $\log_{10}$ transformed p-values. Fig. 2.4c and Supp. Figs. 2.21–2.23c show that Clipper's contrast scores have higher reproducibility by all three criteria compared to edgeR's and DESeq2's p-values.

Finally, we compared Clipper with DESeq2 and edgeR on the real RNA-seq data of classical and non-classical human monocytes [61]. In this dataset, gene expression changes are expected to be associated with the immune response process. We input three classical and three non-classical samples into Clipper, DESeq2, and edgeR for DEG identification. Fig. 2.5a shows that edgeR identifies the fewest DEGs, while DESeq2 identifies the most DEGs, followed by Clipper. Notably, most DEGs identified by DESeq2 are not identified by Clipper or edgeR. To investigate whether DESeq2 makes too many false discoveries and whether the DEGs found by Clipper but missed by DESeq2 or edgeR are biologically meaningful, we performed functional analysis on the set of DEGs identified by each method. We first performed the gene ontology (GO) analysis on the three sets of identified DEGs using the R package `clusterProfiler` [65]. Fig. 2.5b ("Total") shows that more GO terms are enriched (with enrichment q-values $\leqslant 0.01$) in the DEGs identified by Clipper than in the DEGs identified by DESeq2 or edgeR. For the GO terms enriched in all three sets of identified DEGs, Fig. 2.5c shows that they are all related to the immune response and thus biologically meaningful. Notably, these biologically meaningful GO terms have more significant enrichment in the DEGs identified by Clipper than in those identified by edgeR and DESeq2. We further performed GO analysis on the DEGs uniquely identified by one method in pairwise comparisons of Clipper vs. DESeq2 and Clipper vs. edgeR. Fig. 2.5b and Supp. Fig. 2.26 show that multiple immune-related GO terms are enriched in Clipper-specific DEGs, while no GO terms are enriched in edgeR-specific or DESeq2-specific DEGs. In addition, we examined the DEGs that were identified by Clipper only but missed by both edgeR and DESeq2. Fig. 2.5d and Supplementary Table show that these genes include multiple key immune-related genes, including *CD36*, *DUSP2*, and *TNFAIP3*. We further performed pathway analysis on these genes and the DEGs that were identified by DEseq2 only but missed by both edgeR and Clipper, using the R package `limma` [34]. Supp. Fig. 2.27a shows that the DEGs that were only identified by Clipper have significant enrichment for immune-related pathways including phagosome, a key function of monocytes and macrophages. On the contrary, Supp. Fig. 2.27b shows that fewer immune-related pathways are enriched in DEGs that were only identified by DESeq2. Altogether, these results confirm the capacity of Clipper in real-data DEG analysis, and they are consistent with our simulation results that edgeR lacks power, while DESeq2 fails to control the FDR.

DEG identification from single-cell RNA-seq data (differential analysis II)

Single-cell RNA sequencing (scRNA-seq) technologies have revolutionized biomedical sciences by enabling genome-wide profiling of gene expression levels at an unprecedented single-cell resolution. DEG analysis is widely applied to scRNA-seq data for discovering genes whose expression levels change between two conditions or between two cell types. Compared with bulk RNA-seq data, scRNA-seq data have many more "replicates" (i.e., cells, whose number is often in hundreds) under each condition or within each cell type.

We compared Clipper with edgeR [28], MAST [66], Monocle3 [67], the two-sample t

test, and the Wilcoxon rank-sum test, five methods that are either popular or reported to have comparatively top performance from a previous benchmark study [68]. To verify the FDR control, we used scDesign2, a flexible probabilistic simulator to generate scRNA-seq count data with known true DEGs [69]. scDesign2 offers three key advantages that enable the generation of realistic synthetic scRNA-seq count data: (1) it captures distinct marginal distributions of different genes; (2) it preserves gene-gene correlations; (3) it adapts to various scRNA-seq protocols. Using scDesign2, we generated two synthetic scRNA-seq datasets from two real scRNA-seq datasets of peripheral blood mononuclear cells (PBMCs) [70]: one using 10x Genomics [71] and the other using Drop-seq [72]. Each synthetic dataset contains two cell types, CD4+ T cells and cytotoxic T cells, which we treated as two conditions. Having true DEGs known, the synthetic datasets allow us to evaluate Clipper's and the other five methods' FDRs and power for a range of target FDR thresholds: $q = 1\%, 2\%, \ldots, 10\%$. Fig. 2.6d and Supp. Fig. 2.25 show that on both 10x Genomics and Drop-seq synthetic datasets, Clipper consistently controls the FDR and remains the most powerful among all the methods that achieve FDR control. These results demonstrate Clipper's robust performance in scRNA-seq DEG analysis.

Discussion

In this chapter, we proposed a new statistical framework, Clipper, for identifying interesting features with FDR control from high-throughput data. Clipper avoids the use of p-values and makes FDR control more reliable and flexible. We used comprehensive simulation studies to verify the FDR control by Clipper under various settings. We demonstrate that Clipper outperforms existing generic FDR control methods by having higher power and greater robustness to model misspecification. We further applied Clipper to two popular bioinformatics analyses: peak calling from ChIP-seq data, and DEG identification from RNA-seq data. Our results indicate that Clipper can provide a powerful add-on to existing bioinformatics tools to improve the reliability of FDR control and thus the reproducibility of scientific discoveries.

Clipper's FDR control procedures (BC and GZ procedures in Methods) are motivated by the Barber-Candès (BC)'s knockoff paper [50] and the Gimenez-Zou's multiple knockoff paper [54], but we do not need to construct knockoffs in enrichment analysis when two conditions have the same number of replicates; the reason is that the replicates under the background condition serve as natural negative controls. For differential analysis and enrichment analysis with unequal numbers of replicates, in order to guarantee the theoretical assumptions for FDR control, Clipper uses permutations instead of the complicated knockoff construction because Clipper only examines features marginally and does not concern about features' joint distribution.

We validated the FDR control by Clipper using extensive and concrete simulations, including both model-based and real-data-based data generation with ground truths, which are widely used to validate newly developed computational frameworks [73]. In contrast, in most bioinformatics method papers, the FDR control was merely mentioned but rarely validated. Many of them assumed that using the BH procedure on p-values would lead to valid FDR control; however, the reality is often otherwise because p-values would be invalid when model assumptions were violated or the p-value calculation was problematic. Here we voice the importance of validating the FDR control in bioinformatics method development, and we use this work as a demonstration. We believe that Clipper provides a powerful booster to this movement. As a p-value-free alternative to the classic p-value-based BH procedure, Clipper relies less on model assumptions and is thus more robust to model misspecifications, making it an appealing choice for FDR control in diverse high-throughput biomedical data analyses.

Clipper is a flexible framework that is easily generalizable to identify a variety of interesting features. The core component of Clipper summarizes each feature's measurements under each condition into an informative statistic (e.g., the sample mean); then Clipper combines each feature's informative statistics under two conditions into a contrast score to enable FDR control. The current implementation of Clipper only uses the sample mean as the informative statistic to identify the interesting features that have distinct expected values under two conditions. However, by modifying the informative statistic, we can generalize Clipper to identify the features that are interesting in other aspects, e.g., having different variances between two conditions. Regarding the contrast score, Clipper currently makes careful choices between two contrast scores, minus and maximum, based on the number of replicates and the analysis task (enrichment or differential).

Notably, Clipper achieves FDR control and high power using those two simple contrast scores, which are calculated for individual features without borrowing information from other features. However, Clipper does leverage the power of multiple testing by setting a contrast score threshold based on all features' contrast scores. This is a likely reason why Clipper achieves good power even with simple contrast scores. An advantage of Clipper is that it allows other definitions of contrast scores, such as the two-sample t statistic that considers within-condition variances. Empirical evidence (Supp. Figs. 2.19 and 2.20) shows that the Clipper variant using the two-sample t statistic is underpowered by the default Clipper, which uses the minus summary statistic (difference of two conditions' sample means) as the contrast score in the 3vs3 enrichment analysis or as the degree of interestingness in the 3vs3 differential analysis (see Methods). Here is our current interpretation of this seemingly counter-intuitive result.

- First, both the minus statistic and the t statistic satisfy Clipper's theoretical conditions, which guarantee the FDR control by the BC and GZ procedures; this is confirmed in Supp. Figs. 2.19 and 2.20. Hence, from the FDR control perspective, Clipper does not require the adjustment for within-condition variances by using a t statistic.

- Second, Clipper is different from the two-sample t test or the regression-based t test, where the t statistic was purposely derived as a pivotal statistic so that its null distribution (the t distribution) does not depend on unknown parameters. Since Clipper does not require a null distribution for each feature, the advantage of the t statistic being pivotal no longer matters.

- Third, the minus statistic only requires estimates of two conditions' mean parameters, while the t statistic additionally requires estimates of the two conditions' variances. Hence, when the sample sizes (i.e., the numbers of replicates) are small, the two more parameters that need estimation in the t statistic might contribute to the observed power loss of the Clipper t statistic variant. Indeed, the power difference between the two statistics diminishes as the sample sizes increase from 3vs3 in Supp. Figs. 2.19–2.20 to 10vs10 in Supp. Figs. 2.16 (where we compared the default Clipper with BH-pair-parametric, which is based on the two-sample t test and is highly similar to the Clipper t statistic variant).

- Fourth, we observe empirically that a contrast score would have better power if its distribution (based on its values of all features) has a larger range and a heavier right tail (in the positive domain). Compared to the minus statistic, the t statistic has a smaller range and a lighter right tail due to its adjustment for features' within-condition variances (Supp. Fig. 2.34). This observation is consistent with the power difference of the two statistics.

Beyond our current interpretation, however, we admit that future studies are needed to explore alternative contrast scores and their power with respect to data characteristics and analysis tasks. Furthermore, we may generalize Clipper to be robust against sample batch effects by constructing the contrast score as a regression-based test statistic that has batch effects removed.

Our current version of Clipper allows the identification of interesting features between two conditions. However, there is a growing need to generalize our framework to identify features across more than two conditions. For example, temporal analysis based on scRNA-seq data aims to identify genes whose expression levels change along time [46]. To tailor Clipper for such analysis, we could define a new contrast score that differentiates the genes with stationary expression (uninteresting features) from the other genes with varying expression (interesting features). Further studies are needed to explore the possibility of extending Clipper to the regression framework so that Clipper can accommodate data of multiple conditions or even continuous conditions, as well as adjusting for confounding covariates.

We have demonstrated the broad application potential of Clipper in various bioinformatics data analyses. Specifically, when used as an add-on to established, popular bioinformatics methods such as MACS2 for peak calling, Clipper guaranteed the desired FDR control and in some cases boosted the power. However, many more careful thoughts are needed to escalate Clipper into standalone bioinformatics methods for specific data analyses, for which data processing and characteristics (e.g., peak lengths, GC contents, proportions of zeros, and batch effects) must be appropriately accounted for before Clipper is used for the FDR control [68, 74]. We expect that the Clipper framework will propel future development of bioinformatics methods by providing a flexible p-value-free approach to control the FDR, thus improving the reliability of scientific discoveries.

After finishing this manuscript, we were informed of the work by He et al. [75], which is highly similar to the part of Clipper for differential analysis, as both work use permutation for generating negative controls and the GZ procedure for thresholding (test statistics in He et al. and contrast scores in Clipper). However, the test statistics used in He et al. are the two-sample t statistic and the two-sample Wilcoxon statistic, both of which are different from the minus and maximum contrast scores used in Clipper. While we leave the optimization of contrast scores to future work, we note that the two-sample Wilcoxon statistic, though being a valid contrast score for differential analysis, requires a large sample size to achieve good power. For this reason, we did not consider it as a contrast score in the current Clipper implementation, whose focus is on sample-sample-size high-throughout biological data.

2.4 Acknowledgments

This chapter is based on my joint work with Yiling Chen, Dongyuan Song, MeiLu McDermott, Kyla Woyshner, Antigoni Manousopoulou, Dr. Wei Li, and my Ph.D. advisor Dr. Jingyi Jessica Li [13]. We would like to thank Dr. Yu-Cheng Yang for his suggestions on the figures and R package. The authors would also like to thank Mr. Nikos Ignatiadis, Dr. Lihua Lei, and Dr. Rina Barber for their insightful comments after we presented this work at the International Seminar on Selective Inference (https://www.selectiveinferenceseminar.com/past-talks). We also appreciate the comments and feedback from Mr. Tianyi Sun, Ms. Kexin Li, and other members of the Junction of Statistics and Biology at UCLA (http://jsb.ucla.edu).

2.5 Supplementary Material

S2.5.1 Review of generic FDR control methods

To facilitate our discussion, we introduce the notations for data. For feature $j = 1, \ldots, d$, we use $\boldsymbol{X}_j = (X_{j1}, \ldots, X_{jm})^\top \in \mathbb{R}^m$ and $\boldsymbol{Y}_j = (Y_{j1}, \ldots, Y_{jn})^\top \in \mathbb{R}^n$ to denote its measurements under the experimental and background conditions, respectively. We assume that $X_{j1}, \ldots, X_{jm}$ are identically distributed, so are $Y_{j1}, \ldots, Y_{jn}$. Let $\mu_{Xj} = \mathbb{E}[X_{j1}]$ and $\mu_{Yj} = \mathbb{E}[Y_{j1}]$ denote the expected measurement of feature j under the two conditions, respectively. Then we denote by $\bar{X}_j$ the sample average of $X_{j1}, \cdots, X_{jm}$ and by $\bar{Y}_j$ the sample average of $Y_{j1}, \cdots, Y_{jn}$.

S2.5.1.1 P-value-based methods

Here we describe the details of p-value-based FDR control methods, including BH-pair, BH-pool, qvalue-pair, and qvalue-pool. Each of these four methods first computes p-values using either the pooled approach or the paired approach, and it then relies on the BH procedure [1] or Storey's qvalue procedure [2] for FDR control. In short, every p-value-based method is a combination of a p-value calculation approach and a p-value thresholding procedure. Below we introduce two p-value calculation approaches (paired and pooled) and two p-value thresholding procedures (BH and Storey's qvalue).

P-value calculation approaches

The paired approach. The paired approach examines one feature at a time and compares its measurements between two conditions. Besides the ideal implementation, i.e., the *correct paired approach* that uses the correct model to calculate p-values, we also include commonly-used flawed implementations that either misspecify the distribution, i.e., the *misspecified paired approach*, or misformulate the two-sample test as a one-sample test, i.e., the *2as1 paired approach.*

Here we use the negative binomial distribution as an example to demonstrate the ideas of the correct, misspecified, and 2as1 paried approaches. Suppose that for each feature j, its measurements under each condition follow a negative binomial distribution, and the two distributions under the two conditions have the same dispersion; that is, $X_{j1}, \cdots, X_{jm} \overset{\text{i.i.d.}}{\sim} \text{NB}\left(\mu_{Xj}, \theta_j\right)$; $Y_{j1}, \cdots, Y_{jn} \overset{\text{i.i.d.}}{\sim} \text{NB}\left(\mu_{Yj}, \theta_j\right)$, where θ_j is the dispersion parameter such that the variance $\text{Var}(X_{ji}) = \mu_{Xj} + \theta_j \mu_{Xj}^2$.

- The correct paired approach assumes that the two negative binomial distributions have the same dispersion parameter θ_j, and it uses the two-sample test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ (enrichment analysis) or $H_1 : \mu_{Xj} \neq \mu_{Yj}$ (differential analysis).
- The misspecified paired approach misspecifies the negative binomial distribution as Poisson, and it uses the two-sample test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ (enrichment analysis) or $H_1 : \mu_{Xj} \neq \mu_{Yj}$ (differential analysis).
- The 2as1 paired approach bluntly assumes $\mu_{Yj} = \bar{Y}_j$, and it performs the one-sample test based on $X_{j1}, \ldots, X_{jm}$ for the null hypotheses $H_0 : \mu_{Xj} = \bar{Y}_j$ against the alternative hypothesis $H_1 : \mu_{Xj} > \bar{Y}_j$ (enrichment analysis) or $H_1 : \mu_{Xj} \neq \bar{Y}_j$ (differential analysis).

The pooled approach. The pooled approach pools all features' average measurements under the background condition $\left\{\bar{Y}_j\right\}_{j=1}^d$ to form a null distribution, and it calculates a p-value for each feature j by comparing $\bar{X}_j$ to the null distribution. Specifically, in enrichment analysis, the p-value of feature j is computed as:

$$p_j = \frac{\text{card}\left(\left\{k : \bar{Y}_k \geqslant \bar{X}_j\right\}\right)}{d}.$$

In differential analysis, the p-value of feature j is computed as:

$$p_j = 2 \cdot \min\left(\frac{\text{card}\left(\left\{k : \bar{Y}_k \geqslant \bar{X}_j\right\}\right)}{d}, \frac{\text{card}\left(\left\{k : \bar{Y}_k \leqslant \bar{X}_j\right\}\right)}{d}\right).$$

P-value thresholding procedures for FDR control

Definition S2.4 (BH procedure for thresholding p-values [1]). *The features' p-values $p_1, \ldots, p_d$ are sorted in an ascending order $p_{(1)} \leqslant p_{(2)} \leqslant \ldots \leqslant p_{(d)}$. Given the target FDR threshold q, the Benjamini–Hochberg (BH) procedure finds a p-value cutoff T^{BH} as*

$$T^{BH} := p_{(k)}, \text{ where } k = \max\left\{j = 1, \ldots, d : p_{(j)} \leqslant \frac{j}{d} q\right\}. \tag{S2.13}$$

Then BH outputs $\left\{j : p_j \leqslant T^{BH}\right\}$ as discoveries.

Definition S2.5 (Storey's qvalue procedure for thresholding p-values [2]). *The features' p-values $p_1, \ldots, p_d$ are sorted in an ascending order $p_{(1)} \leqslant p_{(2)} \leqslant \ldots \leqslant p_{(d)}$. Let $\hat{\pi}_0$ denote an estimate of the probability $P(\text{the } i\text{-th feature is uninteresting})$ (see Storey [2] for details). Storey's qvalue procedure defines the q-value for $p_{(d)}$ as*

$$\hat{q}(p_{(d)}) := \frac{\hat{\pi}_0 \cdot d \cdot p_{(d)}}{\operatorname{card}\left(\left\{k : p_k \leqslant p_{(d)}\right\}\right)} = \hat{\pi}_0 \cdot p_{(d)} .$$

Then for $j = d-1, d-2, \ldots, 1$, the q-value for $p_{(j)}$ is defined as:

$$\hat{q}(p_{(j)}) := \min\left(\hat{q}(p_{(j+1)}), \frac{\hat{\pi}_0 \cdot d \cdot p_{(j)}}{\operatorname{card}\left(\left\{k : p_k \leqslant p_{(j)}\right\}\right)}\right).$$

Then Storey's qvalue procedure outputs $\{j : \hat{q}(p_j) \leqslant q\}$ as discoveries.

We use function `qvalue` from `R` package `qvalue` (v 2.20.0; with default estimate $\hat{\pi}_0$) to calculate q-values.

Definition S2.6 (SeqStep+ procedure for thresholding p-values [3]). *Define H_0^j as the null hypothesis for feature j and p_j as the p-value for H_0^j, $j = 1, \ldots, d$. Order the null hypotheses $H_0^1, \ldots, H_0^d$ from the most to the least promising (here more promising means more likely to be interesting) and denote the resulting null hypotheses and p-values as $H_0^{(1)}, \ldots, H_0^{(d)}$ and $p_{(1)}, \ldots, p_{(d)}$. Given any target FDR threshold q, a pre-specified constant $s \in (0,1)$, and* *subset $\mathcal{K} \subseteq \{1, \dots, d\}$, the SeqStep+ procedure finds a cutoff $\hat{j}$ as*

$$\hat{j} := \max\left\{ j \in \mathcal{K} : \frac{1 + \mathrm{card}\left(\left\{k \in \mathcal{K}, k \leqslant j : p_{(k)} > s\right)\right\}}{\mathrm{card}\left(\left\{k \in \mathcal{K}, k \leqslant j : p_{(k)} \leqslant s\right)\right\} \vee 1} \leqslant \frac{1-s}{s} q \right\} \tag{S2.14}$$

Then SeqStep+ rejects $\left\{H_0^{(j)} : p_{(j)} \leqslant s,\ j \leqslant \hat{j},\ j \in \mathcal{K}\right\}$. If the orders of the null hypotheses are independent of the p-values, the SeqStep+ procedure ensures FDR control.

The GZ procedure (Definition 3) used in Clipper is a special case of the SeqStep+ procedure with $s = 1/(h+1)$. Recall that given the number of non-identical permutations $h \in \{1, \cdots, h_{\max}\}$ and contrast scores $\{C_j\}_{j=1}^d$, the GZ procedure sorts $\{|C_j|\}_{j=1}^d$ in a decreasing order:

$$|C_{(1)}| \geqslant |C_{(2)}| \geqslant \cdots \geqslant |C_{(d)}|\,. \tag{S2.15}$$

To see the connection between the GZ procedure and SeqStep+, we consider the null hypothesis for the j-th ordered feature, $j = 1, \dots, d$, as $H_0^{(j)} : \mu_{X(j)} = \mu_{Y(j)}$ and define the corresponding p-value $p_{(j)} := \frac{r(T_{(j)}^{\sigma_0})}{h+1}$, where $r(T_{(j)}^{\sigma_0})$ is the rank of $T_{(j)}^{\sigma_0}$ in $\{T_{(j)}^{\sigma_0}, \cdots, T_{(j)}^{\sigma_h}\}$ in a descending order. We also define $\mathcal{K} := \{j = 1, \dots, d : C_j \neq 0\}$ as the subset of features with non-zero C_j's. Finally, we input the p-values, null hypothesis orders in (S2.15), $s = 1/(h+1)$, q and $\mathcal{K}$ into the SeqStep+ procedure, and we obtain the GZ procedure.

The BC procedure (Definition 1) is a further special case with $h = 1$, $p_{(j)} := \left(\mathbb{1}(C_{(j)} > 0) + 1\right)/2$, and $\mathcal{K} := \{j = 1, \dots, d : C_j \neq 0\}$.

S2.5.1.2 Local-fdr-based methods

The FDR is statistical criterion that ensures the reliability of discoveries as a whole. In contrast, the local fdr focuses on the reliability of each discovery. The definition of the local fdr relies on some pre-computed summary statistics z_j for feature j, $j = 1, \dots, d$. In the calculation of local fdr, $\{z_1, \dots, z_d\}$ are assumed to be realizations of an abstract random variable Z that represents any feature. Let p_0 or p_1 denote the prior probability that any feature is uninteresting or interesting, with $p_0 + p_1 = 1$. Let $f_0(z) := \mathbb{P}(Z = z \mid \text{uninteresting})$

or $f_1(z) := \mathbb{P}(Z = z \mid \text{interesting})$ denote the conditional probability density of Z at z given that Z represents an uninteresting or interesting feature. Thus by Bayes' theorem, the posterior probability of any feature being uninteresting given its summary statistic $Z = z$ is

$$\mathbb{P}(\text{uninteresting} \mid Z = z) = p_0 f_0(z)/f(z)\,, \tag{S2.16}$$

where $f(z) := p_0 f_0(z) + p_1 f_1(z)$ is the marginal probability density of Z. Accordingly, the local fdr of feature j is defined as follows.

Definition S2.7 (Local fdr [4])**.** *Given notations defined above, the local fdr of feature j is defined as*

$$\textit{local-fdr}_j := f_0(z_j)/f(z_j)\,.$$

Because $p_0 \leqslant 1$, local-fdr$_j$ is an upper bound of the posterior probability of feature j being uninteresting given its summary statistic z_j, defined in (S2.16)*.*

Note that another definition of the local fdr is the posterior probability $\mathbb{P}(\text{uninteresting} \mid z)$ in (S2.16) [5]. Although this other definition is more reasonable, we do not use it but choose Definition S2.7 because the estimation of p_0 is ususally difficult. Another reason is that uninteresting features are the dominant majority in high-throughput biological data, so p_0 is often close to 1.

We define local-fdr-based methods as a type of FDR control methods by thresholding local fdrs of features under the target FDR threshold q. Although the local fdr is different from FDR, it has been shown that thresholding the local fdrs at q will approximately control the FDR under q [4]. This makes local-fdr-based methods competitors against Clipper and p-value-based methods.

Every local-fdr-based method is a combination of a local fdr calculation approach and a local fdr thresholding procedure. Below we introduce two local fdr calculation approaches (empirical null and swapping) and one local fdr thresholding procedure. After the combination, we have two local-fdr-based methods: locfdr-emp and locfdr-swap.

Local fdr calculation approaches

With $z_1, \ldots, z_d$, the calculation of local fdr defined in Definition S2.7 requires the estimation of f_0 and f, two probability densities. f is estimated by nonparametric density estimation, and f_0 is estimated by either the empirical null approach [4] or the swapping approach, which shuffles replicates between conditions [5]. With the estimated $\hat{f}$ and $\hat{f}_0$, the estimated local fdr of feature j is

$$\widehat{\text{local-fdr}}_j := \hat{f}_0(z_j)/\hat{f}(z_j)\,. \tag{S2.17}$$

The empirical null approach. This approach assumes a parametric distribution, typically the Gaussian distribution, to estimate f_0. Then with the density estimate $\hat{f}$, the local fdr is estimated for each feature j. The implementation of this approach depends on the numbers of replicates.

- In 1vs1 enrichment and differential analyses, we define z_j as

  $$z_j := \frac{D_j}{\sqrt{\frac{1}{d}\sum_{j=1}^{d}\left(D_j - \bar{D}\right)^2}}\,,$$

  where $D_j = X_{j1} - Y_{j1}$ and $\bar{D} = \sum_{j=1}^{d} D_j/d$.

- In 2vs1 enrichment and differential analyses, we define z_j as

  $$z_j := \frac{\bar{X}_j - Y_{j1}}{\sqrt{\frac{s_{Xj}^2}{2}}}\,,$$

  where $s_{Xj}^2 = \sum_{i=1}^{2}(X_{ji} - \bar{X}_j)^2$.

- In $m$vs$n$ enrichment and differential analyses with $m, n \geqslant 2$, we define z_j as the two-sample t-statistic with unequal variances:

  $$z_j := \frac{\bar{X}_j - \bar{Y}_j}{\sqrt{\frac{s_{Xj}^2}{m} + \frac{s_{Yj}^2}{n}}}\,,$$

where $s_{Xj}^2 = \frac{1}{m-1}\sum_{i=1}^m (X_{ji} - \bar{X}_j)^2$ and $s_{Yj}^2 = \frac{1}{n-1}\sum_{i=1}^n (Y_{ji} - \bar{Y}_j)^2$ are the sample variances of feature j under the experimental and background conditions.

Then $\{\widehat{\text{locfdr}}_j\}_{j=1}^d$ are estimated from $\{z_j\}_{j=1}^d$ by function `locfdr` in `R` package `locfdr` (v 1.1-8; with default arguments).

The swapping approach. This approach swaps $\lceil m/2 \rceil$ replicates under the experimental condition with $\lceil n/2 \rceil$ replicates under the background condition. Then it calculates the summary statistic for each feature on the swapped data, obtaining $z_1', \ldots, z_d'$. Finally, it estimates f_0 and f by applying kernel density estimation to $z_1', \ldots, z_d'$ and $z_1, \ldots, z_d$, respectively (by function `kde` in `R` package `ks`). With $\hat{f}_0$ and $\hat{f}$, $\{\widehat{\text{locfdr}}_j\}_{j=1}^d$ are calculated by Definition S2.7.

The implementation of this approach depends on the numbers of replicates. Below are three special cases included in this work.

- In 1vs1 enrichment and differential analyses, the swapping approach is inapplicable because interesting features would not become uninteresting after the swapping.

- In 2vs1 enrichment and differential analyses, we define z_j and z_j' as

$$\begin{aligned} z_j &= \frac{X_{j1} + X_{j2}}{2} - Y_{j1}\,, \\ z_j' &= \frac{X_{j1} + Y_{j1}}{2} - X_{j2}\,. \end{aligned}$$

- In 3vs3 enrichment and differential analyses with, we define z_j and z_j' as

$$\begin{aligned} z_j &= \frac{X_{j1} + X_{j2}}{2} - \frac{Y_{j1} + Y_{j2}}{2}\,, \\ z_j' &= \frac{X_{j1} + Y_{j1}}{2} - \frac{X_{j2} + Y_{j2}}{2}\,. \end{aligned}$$

Then we apply kernel density estimation to $\{z_j\}_{j=1}^d$ and $\left\{z_j'\right\}_{j=1}^d$ to obtain $\hat{f}$ and $\hat{f}_0$, respectively. By (S2.17), we calculate $\{\widehat{\text{locfdr}}_j\}_{j=1}^d$.

The local fdr thresholding procedure

Definition S2.8 (locfdr procedure)**.** *Given the local fdr estimates* $\{\widehat{local\text{-}fdr}_j\}_{j=1}^d$ *and the target FDR threshold* q*, the locfdr procedure outputs* $\{j = 1, \ldots, d : \widehat{local\text{-}fdr}_j \leqslant q\}$ *as discoveries.*

Generic FDR control methods

In our simulation analysis, we compared Clipper against generic FDR control methods including p-value-based methods and local-fdr-based methods. Briefly, each p-value-based method is a combination of a p-value calculation approach and a p-value thresholding procedure. We use either the "paired" or "pooled" approach (see next paragraph) to calculate p-values of features and then threshold the p-values using the BH procedure (Supp. Definition S2.4) or Storey's qvalue procedure (Supp. Definition S2.5) to make discoveries (Supp. Section S2.5.1.1). As a result, we have four p-value-based methods: BH-pair, BH-pool, qvalue-pair, and qvalue-pool (Fig. 2.1b).

Regarding the existing p-value calculation approaches in bioinformatics tools, we categorize them as "paired" or "pooled." The paired approach has been widely used to detect DEGs and protein-binding sites [6–9]. It examines one feature at a time and compares the feature's measurements between two conditions using a statistical test. In contrast, the pooled approach is popular in proteomics for identifying peptide sequences from MS data [10]. For every feature, it defines a test statistic and estimates a null distribution by pooling all features' observed test statistic values under the background condition. Finally, it calculates a p-value for every feature under the experimental condition based on the feature's observed test statistic and the null distribution.

In parallel to p-value-based methods, local-fdr-based methods estimate local fdrs of features and then threshold the local fdrs using the locfdr procedure (Supp. Definition S2.8) to make discoveries. The estimation of local fdrs takes one of two approaches: (1) empirical null, which is estimated parametrically from the test statistic values that are likely drawn from the null distribution, and (2) swapping null, which is constructed by swapping measurements between experimental and background conditions. The resulting two local-fdr-based-methods are referred to as locfdr-emp and locfdr-swap (Figs. 2.1b and 2.3). Supp. Section S2.5.1 provides a detailed explanation of these generic methods and how we implemented them in this work.

Specific to the p-value-based methods, for the paired approach, besides the ideal implementation that uses the correct model to calculate p-values (BH-pair-correct and qvalue-pair-correct), we also consider common mis-implementations. The first mis-implementations is misspecification of the distribution (BH-pair-mis and qvalue-pair-mis). An example is the detection of protein-binding sites from ChIP-seq data. A common assumption is that ChIP-seq read counts in a genomic region (i.e., a feature) follow the Poisson distribution [6, 7], which implies that the counts have the variance equal to the mean. However, if only two replicates are available, it is impossible to check whether this Poisson distribution is reasonably specified. The second mis-implementation is the misspecification of a two-sample test as a one-sample test (BH-pair-2as1 and qvalue-pair-2as1), which ignores the sampling randomness of replicates under one condition. This issue is implicit but widespread in bioinformatics methods [6, 11].

To summarize, we compared Clipper against the following implementations of generic FDR control methods:

- **BH-pool** or **qvalue-pool**: p-values calculated by the pooled approach and thresholded by the BH or qvalue procedure.

- **BH-pair-correct** or **qvalue-pair-correct**: p-values calculated by the paired approach with the correct model specification and thresholded by the BH or qvalue procedure.

- **BH-pair-mis** or **qvalue-pair-mis**: p-values calculated by the paired approach with a misspecified model and thresholded by the BH or qvalue procedure.

- **BH-pair-2as1** or **qvalue-pair-2as1**: p-values calculated by the paired approach that misformulates a two-sample test as a one-sample test (2as1) and thresholded by the

BH or qvalue procedure.

- **locfdr-emp**: local fdrs calculated by the empirical null approach and thresholded by the locfdr procedure.
- **locfdr-swap**: local fdrs calculated by the swapping approach and thresholded by the locfdr procedure.

S2.5.2 Comparison of Clipper variant algorithms

We compared Clipper variant algorithms applicable to each experimental design. Based on the comparison results, we selected a variant algorithm as the default Clipper implementation for each design.

- **1vs1 enrichment analysis.** Under each of the 12 settings, we compared Clipper-minus-BC, Clipper-minus-aBH, Clipper-max-BC, and Clipper-max-aBH (Section 2.2.1), the only four Clipper variant algorithms applicable to 1vs1 enrichment analysis. The results in Fig. 2.28 show that, regardless of the contrast scores being minus or maximum (max), the BC procedure always guarantees the FDR control under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$. Notably, in terms of power, the two contrast scores consistently have different advantages under the two background scenarios: Clipper-max-BC has higher power under the homogeneous background, while Clipper-minus-BC is more powerful under the heterogeneous background. Considering that the heterogeneous scenario is prevalent in high-throughput biological data, the minus contrast score is preferred. As the power of Clipper-minus-BC drops when q is too small ($q \leqslant 3\%$) and d is not too large ($d = 1000$), we consider the aBH procedure as an alternative to control the FDR. The results show that Clipper-minus-aBH is indeed more powerful when Clipper-minus-BC lacks power; however, Clipper-minus-aBH cannot guarantee the exact FDR control as Clipper-minus-BC does. Therefore, Clipper uses **Clipper-minus-BC** by default in 1vs1 enrichment analysis, and it recommends users to increase q when too few discoveries are made; if users reject this option, then

Clipper would use Clipper-minus-aBH to increase power for the current q.

- **2vs1 enrichment analysis.** Under each of the 6 settings, we compared Clipper-minus-GZ and Clipper-max-GZ (Section 2.2.1), the only two Clipper variant algorithms applicable to 2vs1 enrichment analysis. For either algorithm, we further compared two numbers of permutation equivalence classes: $h = 1$ or 2, where the latter is $h_{\max} = \binom{3}{1} - 1$—the maximum number of equivalence classes that do not include the identity permutation. Note that h is a required input parameter for the GZ procedure. The results in Fig. 2.29 show that, regardless of h and the contrast score definition—maximum (max) or minus, the GZ procedure always guarantees the FDR control under all target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$. In terms of power, Clipper-max-GZ($h = 1$) is consistently more powerful than the other three Clipper variants under all settings. Therefore, Clipper uses **Clipper-max-GZ($h = 1$)** by default in enrichment analysis with unequal numbers of replicates under two conditions.

- **3vs3 enrichment analysis.** Under each of the 12 settings, we compared five Clipper variant algorithms: Clipper-minus-BC, Clipper-minus-aBH, Clipper-max-BC, Clipper-max-aBH, and Clipper-max-GZ (Section 2.2.1). Fig. 2.30 shows the comparison of the first four variants: regardless of the contrast scores being minus or maximum (max), the BC procedure simultaneously guarantees the FDR control and achieves good power under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$. Similar to the results in the 1vs1 enrichment analysis, Clipper-max-BC has higher power under the homogeneous background, while Clipper-minus-BC is more powerful under the heterogeneous background. By the same reasoning—the prevalent heterogeneous scenarios in high-throughput biological data—we prefer the minus contrast score. Unlike the 1vs1 enrichment analysis, here Clipper-minus-BC is consistently as powerful as Clipper-minus-aBH, even when q is small, but Clipper-minus-aBH cannot guarantee the exact FDR control. Therefore, Clipper-minus-BC achieves the overall best performance among the first four Clipper variants. Given that the GZ procedure is also applicable to this setting, we further compared Clipper-minus-BC with Clipper-max-

GZ($h = 1$), the most powerful Clipper variant with the GZ procedure and the default Clipper implementation in the 2vs1 enrichment and differential analyses and the 3vs3 differential analysis. The results in Fig. 2.32 show that while both **Clipper-minus-BC** and Clipper-max-GZ($h = 1$) control the FDR, the former is more powerful. Hence, we will use Clipper-minus-BC as the default when both conditions have more than one and the same number of replicates.

Under the simulation settings from Gaussian distributions, we also compared Clipper-minus-BC with another Clipper variant using the BC procedure and the t statistic as the contrast score (Clipper-t), where the t statistic is from the two-sample t test. Fig. 2.19 shows that, although Clipper-t always guarantees the FDR control under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, it has lower power compared to Clipper-minus-BC, our default Clipper for enrichment analysis with equal numbers of replicates. Based on this result, we did not consider the t statistic as an alternative contrast score for Clipper.

- **2vs1 differential analysis.** Similar to 2vs1 enrichment analysis, under each of the 6 settings, we compared Clipper-minus-GZ and Clipper-max-GZ (Section 2.2.1) with $h = 1$ or 2. The results in Fig. 2.29 show that, regardless of h and the contrast score definition—maximum (max) or minus, the GZ procedure always guarantees the FDR control under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$. Notably, in terms of power, Clipper-minus-GZ($h = 2$) is the most powerful when when q is very small ($q \leqslant 2\%$) under Poisson and negative binomial settings, while Clipper-max-GZ($h = 1$) is the most powerful otherwise. Considering that Clipper-max-GZ($h = 1$) outperforms the other three Clipper variants in most cases, Clipper uses **Clipper-max-GZ($h = 1$)** by default in 2vs1 differential analysis, and it recommends users to use Clipper-minus-GZ($h = 2$) when too few discoveries are made.

- **3vs3 differential analysis.** Under each of the 12 settings, we compared Clipper-minus-GZ, and Clipper-max-GZ (Section 2.2.1) with $h = 1$, 3 or 9, where $h = 9$ is $h_{\max} = \binom{6}{3}/2 - 1$—the maximum number of equivalence classes that do not include the identity permutation. The results in Fig. 2.31 show that, regardless of h and the contrast score definition—maximum (max) or minus, the GZ procedure always guarantees the FDR control under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$. In terms of power, Clipper-max-GZ($h = 1$) is consistently more powerful than the other Clipper variant algorithms under all settings. Therefore, Clipper uses **Clipper-max-GZ($h = 1$)** by default in 3vs3 differential analysis.

Under the simulation settings from Gaussian distributions, we also compared Clipper-max-GZ with another Clipper variant using the GZ procedure and the t statistic to calculate the degree of interestingness (Clipper-t), where the t statistic is from the two-sample t test. Fig. 2.20 shows that, although Clipper-t always guarantees the FDR control under a range of target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, it has lower power compared to Clipper-max-GZ, our default Clipper for differential analysis. Based on this result, we did not consider the t statistic as an alternative contrast scores for Clipper.

In summary, whenever Clipper-minus-BC is applicable (enrichment analysis with equal number of replicates under two conditions), it is chosen as the default Clipper implementation; otherwise, Clipper-max-GZ($h = 1$) is the default.

S2.5.3 Data generation and detailed implementation of the paired approach (a p-value calculation approach) in simulation studies

We describe how we simulated data and how we implemented the paired approach in different simulation settings: 1vs1 enrichment analysis, 2vs1 enrichment analysis, 3vs3 enrichment analysis, 2vs1 differential analysis, and 3vs3 differential analysis, combined with three distribution families (Gaussian, Poisson, and negative binomial) and two background scenarios (homogeneous and heterogeneous). Under some settings, we considered different numbers of features and the existence of outliers.

In each simulation setting, we generated 200 simulated datasets, computed an FDP and an empirical power on each dataset, and averaged the 200 FDPs and 200 empirical powers to approximate the FDR and power, repsectively. For notation simplicity, we use $N(\mu, \sigma^2)$ to denote the Gaussian distribution with mean μ and variance σ^2, Pois(λ) to denote the Poisson distribution with mean λ, and NB(μ, θ) to denote the negative binomial distribution with mean μ and dispersion θ (such that its variance equals $\mu + \theta\mu^2$).

For each design and analysis, we compared the default Clipper implementation with other generic FDR control methods. Specifically, seven generic methods (BH-pool, qvalue-pool, BH-pair-mis, qvalue-pair-mis, BH-pair-2as1, qvalue-pair-2as1, and locfdr-emp) are included in all designs and analyses. The two methods relying on correct model specification, BH-pair-correct and qvalue-pair-correct, are only included in the 3vs3 enrichment and differential analyses, because it is almost impossible to correctly specify a model with fewer than three replicates per condition. The permutation-based method, locfdr-swap, is excluded from the 1vs1 enrichment analysis because it requires at least one condition to have more than one replicate.

In addition to the above designs and analyses, we also compared the default Clipper implementation with BH-pair methods that use parametric or non-parametric tests to calculate p-values when the numbers of replicates are 10 under both conditions for enrichment analysis, i.e., 10vs10 enrichment analysis.

S2.5.3.1 1vs1 enrichment analysis

We simulated data with $d = 1000$ and 10,000 features under two background scenarios and three distributional families—a total of 12 settings. In each setting, 10% of the features are interesting ($\mu_{Xj} > \mu_{Yj}$), and the rest are uninteresting (with $\mu_{Xj} = \mu_{Yj}$). Recall that $\mathcal{N}$ denotes the set of uninteresting features.

Gaussian distribution

We simulated data from Gaussian using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 0$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 0$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $N(0, 2^2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.
- We independently generated X_{j1} from $N(\mu_{Xj}, 1)$ and Y_{j1} from $N(\mu_{Yj}, 1)$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that the null distribution of $X_{j1} - Y_{j1}$, $j = 1, \ldots, d$ is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{1}{d-1} \sum_{j=1}^{d} \left(X_{j1} - \frac{1}{d} \sum_{j=1}^{d} X_{j1} \right)^2 + \frac{1}{d-1} \sum_{j=1}^{d} \left(Y_{j1} - \frac{1}{d} \sum_{j=1}^{d} Y_{j1} \right)^2 .$$

This is a misspecified model that assumes that μ_{Xj}'s are all equal and so are μ_{Yj}'s. Then we computed the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $X_{j1} - Y_{j1}$ in $N(0, \hat{\sigma}^2)$, i.e., $1 - \Phi\left(\frac{X_{j1} - Y_{j1}}{\hat{\sigma}}\right)$, where Φ is the cumulative distribution function of $N(0, 1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $N(Y_{j1}, 1)$ conditioning on the observed Y_{j1} as the null distribution of X_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of X_{j1} in $N(Y_{j1}, 1)$, i.e., $1 - \Phi\left(X_{j1} - Y_{j1}\right)$.

Poisson distribution

We simulated data from Poisson using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).
- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from Pois(20). For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).

- We independently generated X_{j1} from $\text{Pois}(\mu_{Xj})$ and Y_{j1} from $\text{Pois}(\mu_{Yj})$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we first defined a log-transformation $f(x) = \log(x + 0.01)$, which we applied to X_{j1} and Y_{j1}, $j = 1, \ldots, d$. We assumed that the null distribution of $f(X_{j1}) - f(Y_{j1})$, $j = 1, \ldots, d$ is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{1}{d-1}\sum_{j=1}^{d}\left(f(X_{j1}) - \frac{1}{d}\sum_{j=1}^{d} f(X_{j1})\right)^2 + \frac{1}{d-1}\sum_{j=1}^{d}\left(f(Y_{j1}) - \frac{1}{d}\sum_{j=1}^{d} f(Y_{j1})\right)^2 .$$

This model misspecifies the Poisson distribution as the log-normal distribution.

Then we computed the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $f(X_{j1}) - f(Y_{j1})$ in $N(0, \hat{\sigma}^2)$, i.e., $1 - \Phi\left(\frac{f(X_{j1}) - f(Y_{j1})}{\hat{\sigma}}\right)$, where Φ is the cumulative distribution function of $N(0, 1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $\text{Pois}(Y_{j1})$ conditioning on the observed Y_{j1} as the null distribution of X_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of X_{j1} in $\text{Pois}(Y_{j1})$, i.e., $\mathbb{P}(Z \geqslant X_{j1})$ where $Z \sim \text{Pois}(Y_{j1})$.

Negative binomial distribution

We simulated data from negative binomial using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $\text{NB}(45, 45^{-1})$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $\text{NB}(20, 20^{-1})$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $\text{NB}(45, 45^{-1})$.

- We independently generated X_{j1} from $\text{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ and Y_{j1} from $\text{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that for each uninteresting feature j, Y_{j1} and X_{j1} follow the same Poisson distribution. We calculated the p-value of feature j from a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ using function `poisson.test` in R package `stats`.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $\mathrm{NB}(Y_{j1}, Y_{j1}^{-1})$ conditioning on the observed Y_{j1} as the null distribution of X_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of X_{j1} in $\mathrm{NB}(Y_{j1}, Y_{j1}^{-1})$.

S2.5.3.2 2vs1 enrichment analysis

We simulated data with $d = 10{,}000$ features under two background scenarios and three distributional families—a total of 6 settings. In each setting, 10% of the features are interesting ($\mu_{Xj} > \mu_{Yj}$) and the rest are uninteresting (with $\mu_{Xj} = \mu_{Yj}$). Recall that $\mathcal{N}$ denotes the set of uninteresting features.

Gaussian distribution

We simulated data from Gaussian using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 0$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 0$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j \in \mathcal{N}}$ i.i.d. from $N(0, 2^2)$ and set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. We next generated $\{\mu_{Yj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(0, 2^2)$ and $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.

- We independently generated X_{j1} and X_{j2} from $N(\mu_{Xj}, 1)$ and Y_{j1} from $N(\mu_{Yj}, 1)$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that the null distribution of $\frac{1}{2}(X_{j1} + X_{j2}) - Y_{j1}$, $j = 1, \ldots, d$, is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{1}{2(2d-1)} \sum_{j=1}^{d} \sum_{i=1}^{2} \left(X_{ji} - \frac{1}{2d} \sum_{j=1}^{d} \sum_{i=1}^{2} X_{ji} \right)^2 + \frac{1}{d-1} \sum_{j=1}^{d} \left(Y_{j1} - \frac{1}{d} \sum_{j=1}^{d} Y_{j1} \right)^2 .$$

This is a misspecified model that assumes μ_{Xj}'s are all equal and so are μ_{Yj}'s. Then we computed the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}$ in $N(0, \hat{\sigma}^2)$, i.e., $1 - \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{\hat{\sigma}}\right)$, where Φ is the cumulative distribution function of $N(0, 1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $N(Y_{j1}, 1/2)$ conditioning on the observed Y_{j1} as the null distribution of $\frac{1}{2}(X_{j1} + X_{j2})$. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $\frac{1}{2}(X_{j1} + X_{j2})$ in $N(Y_{j1}, 1/2)$, i.e., $1 - \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{1/\sqrt{2}}\right)$.

Poisson distribution

We simulated data from Poisson using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from Pois(20). For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).

- We independently generated X_{j1} and X_{j2} from Pois(μ_{Xj}) and Y_{j1} from Pois(μ_{Yj}), $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we first defined a log-transformation $f(x) = \log(x + 0.01)$, which we applied to X_{j1} and Y_{j1}, $j = 1, \ldots, d$. We assumed that the null distribution of $f(X_{j1}) + f(X_{j2}) - 2f(Y_{j1})$, $j = 1, \ldots, d$

is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{6}{d-1} \sum_{j=1}^{d} \left(f(Y_{j1}) - \frac{1}{d} \sum_{j=1}^{d} f(Y_{j1}) \right)^2 .$$

This model misspecifies the Poisson distribution as the log-normal distribution.

Then we computed the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $f(X_{j1}) + f(X_{j2}) - 2f(Y_{j1})$ in $N(0, \hat{\sigma}^2)$, i.e., $1 - \Phi\left(\frac{f(X_{j1})+f(X_{j2})-2f(Y_{j1})}{\hat{\sigma}}\right)$, where Φ is the cumulative distribution function of $N(0,1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we assumed that for each uninteresting feature j, X_{j1} and X_{j2} independently follow Pois(Y_{j1}) conditioning on the observed Y_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ by performing a one-sample Poisson test using the R function `poisson.test` for the null hypothesis $H_0 : \mu_{X_j} = Y_{j1}$ against the alternative hypothesis $H_1 : \mu_{X_j} > Y_{j1}$.

Negative binomial distribution

We simulated data from negative binomial using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from NB$(45, 45^{-1})$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from NB$(20, 20^{-1})$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from NB$(45, 45^{-1})$.

- We independently generated X_{j1} and X_{j2} from NB$(\mu_{Xj}, \mu_{Xj}^{-1})$ and Y_{j1} from NB$(\mu_{Yj}, \mu_{Yj}^{-1})$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that for each uninteresting feature j, X_{ji}, $i = 1, 2$ and Y_{j1} follow the same Poisson distribution. We calculated the p-value of feature j from a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ using the function `poisson.test` in R package `stats`.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $\text{NB}(2Y_{j1}, (2Y_{j1})^{-1})$ conditioning on the observed Y_{j1} as the null distribution of $X_{j1} + X_{j2}$. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $X_{j1} + X_{j2}$ in $\text{NB}(2Y_{j1}, (2Y_{j1})^{-1})$.

S2.5.3.3 3vs3 enrichment analysis

We simulated data with and without outliers under two background scenarios and three distributional families—a total of 12 settings. In each setting, we generated $d = 10{,}000$ features, among which 10% are interesting (with $\mu_{Xj} > \mu_{Yj}$) and the rest are uninteresting (with $\mu_{Xj} = \mu_{Yj}$). For the results in Fig. 2.18, we simulated data without outliers under two background scenarios and three distributional families using two more proportions of interesting features: 20% and 40%. The data generation under the Gaussian, Poisson, and negative binomial distributions is the same as the settings with 10% interesting features.

Under the settings with outliers, we generated $\{O^X_{ji} : j = 1, \ldots, d; i = 1, \ldots, 3\}$ and $\{O^Y_{ji} : j = 1, \ldots, d; i = 1, \ldots, 3\}$ i.i.d. from Bernoulli(0.1), where $O^X_{ji} = 1$ or $O^Y_{ji} = 1$ indicates X_{ji} or Y_{ji} is an outlier, respectively. Under settings without outliers, $O^X_{ji} = O^Y_{ij} = 0$ for all $j = 1, \ldots, d; i = 1, \ldots, 3$.

Gaussian distribution

- Under the homogeneous background scenario, we set $\mu_{Yj} = 0$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 0$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $N(0, 2^2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$.

- We independently generated X_{ji} from $N(\mu_{Xj}, 1)$ if $O^X_{ji} = 0$ or from the top 1% percentile of $N(\mu_{Xj}, 1)$ if $O^X_{ji} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $N(\mu_{Yj}, 1)$ if $O^Y_{ji} = 0$ or from the top 1% percentile of $N(\mu_{Yj}, 1)$ if $O^Y_{ji} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

- For the results in Supp. Fig. 2.19, under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^d$ i.i.d. from $N(0, 2^2)$, and $\{s_j\}_{j=1}^d$ i.i.d. from a uniform distribution $U(0.5, 2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $N(5, 1)$. We then independently generated X_{ji} from $N(\mu_{Xj}, s_j^2)$ if $O^X_{ji} = 0$ or from the top 1% percentile of $N(\mu_{Xj}, s_j^2)$ if $O^X_{ji} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $N(\mu_{Yj}, s_j^2)$ if $O^Y_{ji} = 0$ or from the top 1% percentile of $N(\mu_{Yj}, s_j^2)$ if $O^Y_{ji} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

- For the results in Supp. Fig. 2.17, we generated correlated features. We first selected 10 groups of features (2 groups of interesting features and 8 groups of uninteresting features), with each group containing 200 features. For each group k, we used $k_1, \ldots, k_{200}$ to denote the indices of the 200 features within that group and generated $\{X_{k_l i}\}_{l=1}^{200}$ from a multivariate Gaussian distribution $N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, where $\boldsymbol{\mu}_k = (\mu_{Xk_1}, \ldots, \mu_{Xk_{200}})$ and $\boldsymbol{\Sigma}_k$ is a matrix with diagonal entries as 1 and other entries as a fixed correlation. In our simulation, the fixed correlation took two values: 0.2 and 0.4.

To implement the correct paired approach (as in BH-pair-correct and qvalue-pair-correct), we calculated the p-value of feature j from a two-sample t-test with equal variance for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we calculated the p-value of feature j from a two-sample t-test with unequal variance for the null hypothesis $H_0 : \mu_{X_j} = \mu_{Y_j}$ against the alternative hypothesis $H_1 : \mu_{X_j} > \mu_{Y_j}$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we assumed that for each uninteresting feature j, X_{ji}, $i = 1, \ldots, 3$ are i.i.d. Gaussian with mean $\bar{Y}_j$ conditioning on the observed $\bar{Y}_j$ and unknown variance. We calculated the p-value of feature j using a one-sample t-test for the null hypothesis $H_0 : \mu_{X_j} = \bar{Y}_j$ against the alternative hypothesis $H_1 : \mu_{X_j} > \bar{Y}_j$.

Poisson distribution

We simulated data from Poisson using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).
- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from Pois(20). For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from Pois(40).
- We independently generated X_{ji} from Pois(μ_{Xj}) if $O_{ji}^X = 0$ or from the top 1% percentile of Pois(μ_{Xj}) if $O_{ji}^X = 1$, $j = 1, \ldots, d$, $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from Pois(μ_{Yj}) if $O_{ji}^Y = 0$ or from the top 1% percentile of Pois(μ_{Yj}) if $O_{ji}^Y = 1$, $j = 1, \ldots, d$; $1, \ldots, 3$.

To implement the correct paired approach (as in BH-pair-correct and qvalue-pair-correct), we calculated the p-value of feature j by performing a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ using the function `poisson.test` in R package `stats`.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we first defined a log-transformation $f(x) = \log(x + 0.01)$, which we applied to X_{ji} and Y_{ji}, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. We assumed that for each uninteresting feature j, $\{f(X_{ji})\}_{i=1}^3$ and $\{f(Y_{ji})\}_{i=1}^3$ follow Gaussian distributions with mean $\mu_{f(Xj)}$ and $\mu_{f(Yj)}$, respectively. Then we computed the p-value of feature j using a two-sample equal variance t-test for the null hypothesis $H_0 : \mu_{f(Xj)} = \mu_{f(Yj)}$ against the alternative hypothesis $H_1 : \mu_{f(Xj)} > \mu_{f(Yj)}$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we assumed that for each uninteresting feature j, $\{X_{ji}\}_{i=1}^3$ follow Pois($\bar{Y}_j$) conditioning on the observed $\bar{Y}_j$. We calculated the p-value of feature j by performing a one-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \bar{Y}_j$ against the alternative hypothesis $H_1 : \mu_{Xj} > \bar{Y}_j$ using R function `poisson.test` from package `stats`.

Negative binomial distribution

We simulated data from negative binomial using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $\mathrm{NB}(45, 45^{-1})$.
- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $\mathrm{NB}(20, 20^{-1})$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For interesting features, we generated $\{\mu_{Xj}\}_{j \notin \mathcal{N}}$ i.i.d. from $\mathrm{NB}(45, 45^{-1})$.
- We independently generated X_{ji} from $\mathrm{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ if $O_{ji}^{X} = 0$ or from the top 1% percentile of $\mathrm{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ if $O_{ji}^{X} = 1$, $j = 1, \ldots, d$, $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $\mathrm{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$ if $O_{ji}^{Y} = 0$ or from the top 1% percentile of $\mathrm{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$ if $O_{ji}^{Y} = 1$, $j = 1, \ldots, d$, $i = 1, \ldots, 3$.

To implement the correct paired approach (as in BH-pair-correct and qvalue-pair-correct), we performed a two-sample negative binomial test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative $H_1 : \mu_{Xj} > \mu_{Yj}$ using $T_j := \sum_{i=1}^{3} X_{ji} - \sum_{i=1}^{3} Y_{ji}$ as the test statistic. We computed the p-value of feature j as the right tail probability

$$\mathbb{P}(T_j \geqslant t_j) = \sum_{k_1=0}^{\infty} \sum_{k_2=k_1+t_j}^{\infty} \mathbb{P}\left(\sum_{i=1}^{3} X_{ji} \geqslant k_2\right) \mathbb{P}\left(\sum_{i=1}^{3} Y_{ji} = k_1\right),$$

where t_j is the realization of T_j, $\mathbb{P}(\sum_{i=1}^{3} X_{ji} \geqslant k_2)$ and $\mathbb{P}(\sum_{i=1}^{3} Y_{ji} = k_1)$ can be estimated from the null distribution of X_{ji} and Y_{ji}, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. As $\sum_{i=1}^{3} X_{ji}$ and $\sum_{i=1}^{3} Y_{ji}$ follow the same distribution under null, we estimated μ_{Xj} and μ_{Yj} as $\hat{\mu}_{Xj} = \hat{\mu}_{Yj} := (\sum_{i=1}^{3} X_{ji} + \sum_{i=1}^{3} Y_{ji})/6$. Then, we calculated $\mathbb{P}(\sum_{i=1}^{3} X_{ji} \geqslant k_2)$ and $\mathbb{P}(\sum_{i=1}^{3} Y_{ji} = k_1)$ using the estimated distribution of X_{ji} and Y_{ji} as $\mathrm{NB}(\hat{\mu}_{Xj}, (\hat{\mu}_{Xj})^{-1})$ and $\mathrm{NB}(\hat{\mu}_{Yj}, (\hat{\mu}_{Yj})^{-1})$, respectively, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that for each uninteresting feature j, $\{X_{ji}\}_{j=1}^{3}$ and $\{Y_{ji}\}_{j=1}^{3}$ follow the same Poisson distribution. We calculated the p-value of feature j from a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} > \mu_{Yj}$ using function `poisson.test` in R package `stats`.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $\mathrm{NB}(\sum_{i=1}^{3} Y_{ji}, (\sum_{i=1}^{3} Y_{ji})^{-1})$ conditioning on the observed $\sum_{i=1}^{3} Y_{ji}$ as the null distribution of $\sum_{i=1}^{3} X_{ji}$. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the right tail probability of $\sum_{i=1}^{3} X_{ji}$ in $\mathrm{NB}(\sum_{i=1}^{3} Y_{ji}, (\sum_{i=1}^{3} Y_{ji})^{-1})$.

S2.5.3.4 10vs10 enrichment analysis

We simulated data without outliers under heterogeneous background scenario and three distributional families—a total of 3 settings. In each setting, we generated $d = 10{,}000$ features, among which 10% are interesting (with $\mu_{Xj} > \mu_{Yj}$) and the rest are uninteresting (with $\mu_{Xj} = \mu_{Yj}$).

The data generation under the Gaussian, Poisson, and negative binomial distributions is the same as in the 3vs3 enrichment analysis (Section S2.5.3.3) except that we set the number of replicates to 10 under each condition, and we did not generate outliers.

The correct paired approaches in BH-pair-parametric are the same as the corresponding BH-pair-correct in the 3vs3 enrichment analysis (Section S2.5.3.3) except that, under the negative binomial distribution, the test statistic T_j and its null distribution should have the number of replicates changed from 3 to 10. The misspecified and 2as1 paired approaches (BH-pair-mis and BH-pair-2as1) are also the same as the corresponding approaches in the 3vs3 enrichment analysis (Section S2.5.3.3).

To implement the non-parametric paired approaches, we calculated the p-value of feature j from the one-sided two-sample Wilcoxon rank-sum test (using `R` function `wilcox.test` in package `stats`) in BH-pair-Wilcoxon and from the one-sided two-sample permutation test (using `R` function `oneway_test` in package `coin`) in BH-pair-permutation.

S2.5.3.5 2vs1 differential analysis

We simulated data with $d = 10{,}000$ features under two background scenarios and three distributional families—a total of 6 settings. In each setting, we set 10% features as "up-regulated" with $\mu_{Xj} > \mu_{Yj}$ and another 10% features as "down-regulated" with $\mu_{Xj} < \mu_{Yj}$.

Gaussian distribution

We simulated data from Gaussian using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 0$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 0$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $N(5, 1)$. For down-regulated features, generated μ_{Xj} i.i.d. from $N(-5, 1)$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $N(0, 2^2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $N(5, 1)$. For down-regulated features, generated μ_{Xj} i.i.d. from $N(-5, 1)$.

- We independently generated X_{j1} and X_{j2} from $N(\mu_{Xj}, 1)$ and Y_{j1} from $N(\mu_{Yj}, 1)$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that the null distribution of $\frac{1}{2}(X_{j1} + X_{j2}) - Y_{j1}$, $j = 1, \ldots, d$, is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{1}{2(2d-1)} \sum_{j=1}^{d} \sum_{i=1}^{2} \left(X_{ji} - \frac{1}{2d} \sum_{j=1}^{d} \sum_{i=1}^{2} X_{ji} \right)^2 + \frac{1}{d-1} \sum_{j=1}^{d} \left(Y_{j1} - \frac{1}{d} \sum_{j=1}^{d} Y_{j1} \right)^2 .$$

This is a misspecified model assuming that μ_{Xj}'s are all equal and so are μ_{Yj}'s. Then we computed the p-value of feature $j = 1, \ldots, d$ as the two-sided tail probability of $\frac{1}{2}(X_{j1} +$

$X_{j2}) - Y_{j1}$ in $N(0, \hat{\sigma}^2)$, i.e., $2 \cdot \min\left(1 - \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{\hat{\sigma}}\right), \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{\hat{\sigma}}\right)\right)$, where Φ is the cumulative distribution function of $N(0, 1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $N(Y_{j1}, 1)$ conditioning on the observed Y_{j1} as the null distribution of X_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the two-sided tail probability of $\frac{1}{2}(X_{j1} + X_{j2})$ in $N(Y_{j1}, 1/2)$, i.e., $2 \cdot \min\left(1 - \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{1/\sqrt{2}}\right), \Phi\left(\frac{\frac{1}{2}(X_{j1}+X_{j2})-Y_{j1}}{1/\sqrt{2}}\right)\right)$.

Poisson distribution

We simulated data from Poisson using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from Pois(60). For down-regulated features, we generated μ_{Xj} i.i.d. from Pois(5) .

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from Pois(20). For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from Pois(60). For down-regulated features, we generated μ_{Xj} i.i.d. from Pois(5) .

- We independently generated X_{j1} and X_{j2} from Pois(μ_{Xj}) and Y_{j1} from Pois(μ_{Yj}), $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we first defined a log-transformation $f(x) = \log(x + 0.01)$, which we applied to X_{j1} and Y_{j1}, $j = 1, \ldots, d$. We assumed that the null distribution of $f(X_{j1}) + f(X_{j2}) - 2f(Y_{j1})$, $j = 1, \ldots, d$ is $N(0, \hat{\sigma}^2)$, where

$$\hat{\sigma}^2 = \frac{6}{d-1} \sum_{j=1}^{d} \left(f(Y_{j1}) - \frac{1}{d} \sum_{j=1}^{d} f(Y_{j1}) \right)^2 .$$

This model misspecifies the Poisson distribution as the log-normal distribution. Then we computed the p-value of feature $j = 1, \ldots, d$ as the two-sided tail probability of $f(X_{j1}) +$

$f(X_{j2}) - 2f(Y_{j1})$ in $N(0, \hat{\sigma}^2)$, i.e., $2 \cdot \min\left(1 - \Phi\left(\frac{f(X_{j1})+f(X_{j2})-2f(Y_{j1})}{\hat{\sigma}}\right), \Phi\left(\frac{f(X_{j1})+f(X_{j2})-2f(Y_{j1})}{\hat{\sigma}}\right)\right)$, where Φ is the cumulative distribution function of $N(0,1)$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we assumed that for each uninteresting feature j, X_{j1} and X_{j2} independently follow $\text{Pois}(Y_{j1})$ conditioning on the observed Y_{j1}. Then we calculated the p-value of feature $j = 1, \ldots, d$ by performing a one-sample Poisson test using the R function `poisson.test` for the null hypothesis $H_0 : \mu_{X_j} = Y_{j1}$ against the alternative hypothesis $H_1 : \mu_{X_j} \neq Y_{j1}$.

Negative binomial distribution

We simulated data from negative binomial using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 30$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 30$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(70, 70^{-1})$. For down-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(7, 7^{-1})$.
- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $\text{NB}(30, 30^{-1})$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(70, 70^{-1})$. For down-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(7, 7^{-1})$.
- We independently generated X_{j1} and X_{j2} from $\text{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ and Y_{j1} from $\text{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$, $j = 1, \ldots, d$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that for each uninteresting feature j, X_{j1}, X_{j2}, and Y_{j1} follow the same Poisson distribution. We calculated the p-value of feature j from a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \mu_{Yj}$ using the function `poisson.test` in R package `stats`.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $\text{NB}(2Y_{j1}, (2Y_{j1})^{-1})$ conditioning on the observed Y_{j1} as the null distribution of $X_{j1} +$

X_{j2}. Then we calculated the p-value of feature $j = 1, \ldots, d$ as the two-sided tail probability of $X_{j1} + X_{j2}$ in $\text{NB}(2Y_{j1}, (2Y_{j1})^{-1})$, i.e., twice the smaller of the left-tail and right-tail probabilities.

S2.5.3.6 3vs3 differential analysis

We simulated data with or without outliers under two background scenarios and three distributional families—a total of 12 settings. In each setting, we generated d = 10,000 features, among which 10% features were "up-regulated features" with $\mu_{Xj} > \mu_{Yj}$ and another 10% were "down-regulated features" with $\mu_{Xj} < \mu_{Yj}$.

Under the settings with outliers, we generated $\{O_{ji}^X : j = 1, \ldots, d; i = 1, \ldots, 3\}$ and $\{O_{ji}^Y : j = 1, \ldots, d; i = 1, \ldots, 3\}$ i.i.d. from Bernoulli(0.1), where $O_{ji}^X = 1$ or $O_{ji}^Y = 1$ indicates X_{ji} or Y_{ji} is an outlier, respectively. Under settings without outliers, $O_{ji}^X = O_{ij}^Y = 0$ for all $j = 1, \ldots, d; i = 1, \ldots, 3$.

Gaussian distribution

- Under the homogeneous background scenario, we set $\mu_{Yj} = 0$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 0$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $N(5, 1)$. For down-regulated features, generated μ_{Xj} i.i.d. from $N(-5, 1)$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^d$ i.i.d. from $N(0, 2^2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $N(5, 1)$. For down-regulated features, generated μ_{Xj} i.i.d. from $N(-5, 1)$.

- We independently generated X_{ji} from $N(\mu_{Xj}, 1)$ if $O_{ji}^X = 0$ or from the top 1% percentile of $N(\mu_{Xj}, 1)$ if $O_{ji}^X = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $N(\mu_{Yj}, 1)$ if $O_{ji}^Y = 0$ or from the top 1% percentile of $N(\mu_{Yj}, 1)$ if $O_{ji}^Y = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

- For the results in Supp. Fig. 2.20, under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $N(0, 2^2)$, and $\{s_j\}_{j=1}^{d}$ i.i.d. from a uniform distribution $U(0.5, 2)$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $N(5, 1)$. For down-regulated features, generated μ_{Xj} i.i.d. from $N(-5, 1)$. We then independently generated X_{ji} from $N(\mu_{Xj}, s_j^2)$ if $O_{ji}^{X} = 0$ or from the top 1% percentile of $N(\mu_{Xj}, s_j^2)$ if $O_{ji}^{X} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $N(\mu_{Yj}, s_j^2)$ if $O_{ji}^{Y} = 0$ or from the top 1% percentile of $N(\mu_{Yj}, s_j^2)$ if $O_{ji}^{Y} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

To implement the correct paired approach (as in BH-pair-correct and qvalue-pair-correct), we calculated the p-value of feature j from a two-sample t-test with equal variance for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \mu_{Yj}$.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we calculated the p-value of feature j from a two-sample t-test with unequal variance for the null hypothesis $H_0 : \mu_{X_j} = \mu_{Y_j}$ against the alternative hypothesis $H_1 : \mu_{X_j} \neq \mu_{Y_j}$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we treated $N(\bar{Y}_j, 1)$ conditioning on observed $\bar{Y}_j$ as the null distribution of X_{ji}, $i = 1, \ldots, 3$. We calculated the p-value of feature j using a one-sample t-test for the null hypothesis $H_0 : \mu_{X_j} = \bar{Y}_j$ against the alternative hypothesis $H_1 : \mu_{X_j} \neq \bar{Y}_j$.

Poisson distribution

We simulated data from Poisson using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 20$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 20$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from Pois(40). For down-regulated features, we generated μ_{Xj} i.i.d. from Pois(5).

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from Pois(20). For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from Pois(40). For down-regulated features, we generated μ_{Xj} i.i.d. from Pois(5) .

- We independently generated X_{ji} from Pois(μ_{Xj}) if $O_{ji}^X = 0$ or from the top 1% percentile of Pois(μ_{Xj}) if $O_{ji}^X = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from Pois(μ_{Yj}) if $O_{ji}^Y = 0$ or from the top 1% percentile of Pois(μ_{Yj}) if $O_{ji}^Y = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

To implement the correct paired approach (as in BH-pair-correct and qvalue-pair-correct), we calculated the p-value of feature j by performing a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \mu_{Yj}$ using function `poisson.test` in R package `stats`.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we first defined a log-transformation $f(x) = \log(x + 0.01)$, which we applied to X_{ji} and Y_{ji}, $j = 1, \ldots, d$; $i = 1, \ldots, 3$. We assumed that for each uninteresting feature j, $\{f(X_{ji})\}_{i=1}^3$ and $\{f(Y_{ji})\}_{i=1}^3$ follow Gaussian distributions with mean $\mu_{f(Xj)}$ and $\mu_{f(Yj)}$, respectively. Then we computed the p-value of feature j using a two-sample equal variance t-test for the null hypothesis $H_0 : \mu_{f(Xj)} = \mu_{f(Yj)}$ against the alternative hypothesis $H_1 : \mu_{f(Xj)} \neq \mu_{f(Yj)}$.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we assumed that for each uninteresting feature j, $\{X_{ji}\}_{i=1}^3$ follow Pois($\bar{Y}_j$) conditioning on the observed $\bar{Y}_j$. We calculated the p-value of feature j by performing a one-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \bar{Y}_j$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \bar{Y}_j$ using the function `poisson.test` in R package `stats`.

Negative binomial distribution

We simulated data from negative binomial using the following procedure:

- Under the homogeneous background scenario, we set $\mu_{Yj} = 30$ for all d features. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj} = 30$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from NB$(70, 70^{-1})$. For down-regulated features, we generated

μ_{Xj} i.i.d. from $\text{NB}(7, 7^{-1})$.

- Under the heterogeneous background scenario, we generated $\{\mu_{Yj}\}_{j=1}^{d}$ i.i.d. from $\text{NB}(30, 30^{-1})$. For uninteresting features, we set $\mu_{Xj} = \mu_{Yj}$ for $j \in \mathcal{N}$. For up-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(70, 70^{-1})$. For down-regulated features, we generated μ_{Xj} i.i.d. from $\text{NB}(7, 7^{-1})$.

- We independently generated X_{ji} from $\text{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ if $O_{ji}^{X} = 0$ or from the top 1% percentile of $\text{NB}(\mu_{Xj}, \mu_{Xj}^{-1})$ if $O_{ji}^{X} = 1$; $j = 1, \ldots, d$, $i = 1, \ldots, 3$. Similarly, we independently generated Y_{ji} from $\text{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$ if $O_{ji}^{Y} = 0$ or from the top 1% percentile of $\text{NB}(\mu_{Yj}, \mu_{Yj}^{-1})$ if $O_{ji}^{Y} = 1$, $j = 1, \ldots, d$; $i = 1, \ldots, 3$.

To implement the correct paired approach with unknown dispersion (as in BH-pair-correct and qvalue-pair-correct), we performed a two-sample negative binomial test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \mu_{Yj}$ using the coefficient from the negative binomial regression as the test statistic. Specifically, for each feature j we performed a negative binomial regression by treating the condition labels as a categorical covariate and feature j's measurements as the response. We implemented this regression analysis using function `glm.nb` in R package `MASS` and extracted the p-value of the coefficient as the p-value of feature j. The dispersion parameter was not pre-specified but estimated by `glm.nb`.

To implement the correct paired approach with known dispersion, we performed a similar negative binomial regression but with the pre-specified dispersion parameter 30^{-1} for each feature j. Then we computed the feature's p-value as the p-value of the coefficient of the condition covariate. We implemented this regression analysis using function `glm` in R package `stats`.

To implement the misspecified paired approach (as in BH-pair-mis and qvalue-pair-mis), we assumed that for each uninteresting feature j, $\{X_{ji}\}_{j=1}^{3}$ and $\{Y_{ji}\}_{j=1}^{3}$ follow the same Poisson distribution. We calculated the p-value of feature j from a two-sample Poisson test for the null hypothesis $H_0 : \mu_{Xj} = \mu_{Yj}$ against the alternative hypothesis $H_1 : \mu_{Xj} \neq \mu_{Yj}$ using function `poisson.test` in R package `stats`.

To implement the 2as1 paired approach (as in BH-pair-2as1 and qvalue-pair-2as1), we first used function `glm.nb` in R package `MASS` to estimate $\hat{\mu}_{Yj}$ and $\hat{\theta}_{Yj}$ from $\{Y_{ji}\}_{j=1}^{3}$. Then we computed the p-value of feature j by treating $\mathrm{NB}(3\hat{\mu}_{Yj}, (3\hat{\theta}_{Yj})^{-1})$ as the null distribution of $\sum_{i=1}^{3} X_{ji}$ and calculated its two-sided tail probability, i.e., twice the smaller of the left-tail and right-tail probabilities.

S2.5.4 Bioinformatic methods with FDR control functionality

Peak calling methods for ChIP-seq data

MACS2 MACS2 [6] uses sliding windows with a fixed length across the genome and identifies peaks by using a Poisson distribution to model the read counts within each window, which has one read count per replicate. Specifically, for each region (which is combined from sliding windows), MACS2 performs a one-sample Poisson test to calculate a p-value, where the null distribution is set to be Poisson with its parameter estimated from the background. By thresholding p-values, MACS2 identifies a set of candidate peaks. It also estimates for each candidate peak a q-value by swapping the experimental sample with the background (negative control) sample, and the q-values are used for FDR control. We used MACS2 software (version 2.2.6) with its default settings.

HOMER We used findPeaks, a program in HOMER [7], to perform peak calling on ChIP-seq data. The p-value calculation in findPeaks is similar to that in MACS2; that is, findPeaks also uses the Poisson distribution as the null distribution of read counts in a genomic region, and it also estimates the Poisson mean from the background sample. Then findPeaks identifies peaks by setting thresholds on p-values and fold-changes (the folder change of a region is defined as the observed read count under the experimental sample divided by the estimated Poisson mean from the the background sample). We used findPeaks version 3.1.9.2.

Differentially expressed gene (DEG) methods for bulk RNA-seq data

edgeR edgeR models each gene's read counts by using a negative binomial regression, where the condition is incorporated as an indicator covariate, and the condition's coefficient represents the gene-wise differential expression effect [8]. We used `R` package `edgeR` version 3.30.0.

DESeq2 DESeq2 uses a similar negative binomial regression as edgeR to model each gene's read counts under two conditions. DESeq2 differs from edgeR mainly in their estimation of the dispersion parameter in the negative binomial distribution [9]. We used `R` package `DESeq2` version 1.28.1.

Differentially expressed gene (DEG) methods for scRNA-seq data

MAST MAST models each gene's log read counts (TPM) by using a two-part generalized regression model. Each gene's expression rate was modeled using logistic regression and, conditioning on a cell expressing the gene, the gene's expression level was modeled as Gaussian [12]. We used `R` package `MAST` version 1.14.0.

Monocle3 Monocle3 uses a generalized linear model to model each gene's normalized expression value, with other information included as covariates (time, treatment, and so on) [13]. We used `R` package `monocle3` version 0.2.3.0.

S2.5.5 Benchmark data generation in omics data applications

ChIP-seq data with synthetic spike-in peaks

We used two control samples (which we refer to as Control 1 and Control 2) from H3K4me3 ChIP-seq data in Chromosome 1 of the cell line GM12878 [14].

(i) We created two semi-synthetic experimental samples by adding synthetic true peaks to Control 1. To mimic real H3K4me3 ChIP-seq data, where peaks are located predominantly in promoter regions, we added synthetic true peaks to promoter regions annotated from Ensembl BioMart (Ensemble hg 19, regulation 104) [15]. Specifically, we randomly sampled 585 genes' promoter regions from Chromosome 1. We then used ChIPulate to simulate reads from these promoter regions (for each simulation, extraction efficiency parameter and PCR efficiency parameter were randomly sampled from a uniform distribution between 0 to 1; binding energy parameters were randomly sampled from a uniform distribution between 0 and 2; sequencing depth parameter was set to 50). Then we added the simulated reads to Control 1. We repeated this procedure for twice to obtain two semi-synthetic experimental samples (i.e., two replicates under the experimental condition).

(ii) We repeated Step (i) for 20 times to generate 20 sets of semi-synthetic experimental samples. For each set of experimental samples, we paired them with Control 2, which was treated as the background sample (i.e., one replicate under the background condition). Hence, we obtained 20 semi-synthetic ChIP-seq datasets, each containing 585 synthetic true peaks.

(iii) After applying a peak calling method to these 20 semi-synthetic datasets, we evaluated the method's 20 FDPs and 20 empirical power, which were then averaged as the method's approximate FDR and power. In the evaluation, a called peak was a true positive if it overlapped with a synthetic true peak; otherwise, it was a false positive.

Bulk RNA-seq data with synthetic spike-in DEGs

We generated four sets of realistic semi-synthetic data from two real RNA-seq datasets. The first one is a human monocyte RNA-seq dataset including 17 samples of classical monocytes and 17 samples of non-classical monocytes [16]. Each sample contains expression levels of $d = 52{,}376$ transcripts.

The second one is a yeast RNA-seq dataset including 48 samples of a *snf2* knockout mutant cell line and 48 samples of negative control (without the knockout) [17]. Each sample contains expression levels of $d = 7126$ genes. We preprocessed this dataset by removing lowquality replicates (replicates 6, 13, 25, 35 from the knockout; replicates 21, 22, 25, 28, 34, 36 from the control) identified by the original paper Gierliński et al. [17], leaving us with 44 replicates under the knockout condition and 42 replicates under the negative control.

Here we describe our **simulation strategy 1**. Given either the human monocyte dataset or the yeast dataset, we performed the following steps.

(i) We first performed normalization on all samples across two conditions using the edgeR normalization method trimmed mean of M-values (TMM) [18]. We denote the resulting normalized read count matrix of classical human monocytes or yeasts without the knockout by $\mathbf{X}^1$ and the normalized read count matrix of non-classical human monocytes or yeast with the knockout by $\mathbf{X}^2$, respectively. Following the convention in bioinformatics, the columns and rows of $\mathbf{X}^1$ and $\mathbf{X}^2$ represent biological samples and genes, respectively.

(ii) To define true DEGs, we first computed the fold change of gene j by $\mathrm{FC}_j = \left[(\bar{\mathbf{X}}_j^2 + 1)/(\bar{\mathbf{X}}_j^1 + 1)\right]$ for $j = 1, \ldots, d$, where $\mathbf{X}_j^1$ and $\mathbf{X}_j^2$ denote the j-th row vector of $\mathbf{X}^1$ and $\mathbf{X}^2$ respectively and $\bar{\cdot}$ denotes the average of elements in a vector. We added the pseudo-count of 1 to avoid division by 0. We defined true DEGs as those with $|\log_2 \mathrm{FC}_j| \geqslant 4$ for the human monocyte dataset and with $|\log_2 \mathrm{FC}_j| \geqslant 1.5$ for the yeast dataset, resulting 191 true human DEGs (transcripts) and 152 true yeast DEGs.

(iii) We generated semi-synthetic data with 3 samples under both the experimental and background conditions, a typical design in bulk RNA-seq experiments. Specifically, if gene j is a true DEG, we randomly sampled without replacement 3 values from $\mathbf{X}_j^1$ as counts under the experimental condition, and another 3 values from $\mathbf{X}_j^2$ as counts under the background condition. If gene j is not a true DEG, we randomly sampled 6 values without replacement from $(\mathbf{X}_j^1, \mathbf{X}_j^2)$ and randomly split them into 3 and 3 counts under two conditions. Doing so guaranteed that a non-DEG's read counts are i.i.d. regardless of condition.

(iv) We repeated Step (iii) for 100 times to generate 100 semi-synthetic datasets.

Next, we describe our **simulation strategy 2**. Let us now re-use notations $\mathbf{X}^1$ to denote the original read count matrix of classical human monocytes or yeast without the knockout, and $\mathbf{X}^2$ to denote the original read count matrix of non-classical human monocytes or yeast with the knockout. Both $\mathbf{X}^1$ and $\mathbf{X}^2$ have rows as genes or transcripts and columns as biological samples. Given either the human monocyte dataset or the yeast dataset, we performed the following steps.

(i) We first identified genes whose read counts are positive in all samples under both conditions and denote the number of such genes by d_p. Then from these identified genes, we randomly sampled without replacement $\min(d_p, 0.3d)$ genes as true DEGs. The remaining $d - \min(d_p, 0.3d)$ genes were considered true non-DEGs.

(ii) To generate fold changes of true DEGs, we first computed the fold change of gene j by $\mathrm{FC}_j = \left[(\bar{\mathbf{X}}_j^2 + 1)/(\bar{\mathbf{X}}_j^1 + 1)\right]$ for $j = 1, \ldots, d$, where $\mathbf{X}_j^1$ and $\mathbf{X}_j^2$ denote the j-th row vector of $\mathbf{X}^1$ and $\mathbf{X}^2$ respectively and $\bar{\cdot}$ denotes the average of elements in a vector. Let $\mathcal{W}$ denote $\{\mathrm{FC}_j : \mathrm{FC}_j \geqslant 16, j = 1, \ldots, d\}$ for the human monocyte dataset and $\{\mathrm{FC}_j : \mathrm{FC}_j \geqslant 1.5, j = 1, \ldots, d\}$ for the yeast dataset. We then sorted unique elements in $\mathcal{W}$ and denoted them by $w_{(1)} < \cdots < w_{(n_u)}$, where n_u is the number of unique elements in $\mathcal{W}$. To generate a fold change of a true DEG, say gene j, we randomly generated an integer v with equal probability from $\{1, \cdots, n_u - 1\}$ and a value p from Uniform$(0, 1)$. Then we calculated the fold change as $R_j = w_{(v)} + p(w_{(v+1)} - w_{(v)})$. Using this approach, generated the fold changes independently for all true DEGs.

(iii) Next, we randomly sampled 6 replicates without replacement from $\mathbf{X}^2$ and split them into two groups of 3 replicates. We denote the resulting matrices as $\widetilde{\mathbf{X}}^1$ and $\widetilde{\mathbf{X}}^2$, whose j-th rows are denoted respectively by $\widetilde{\mathbf{X}}_j^1$ and $\widetilde{\mathbf{X}}_j^2$. If gene j is a true DEG, we generated U_j from Bernoulli(1/2). Then we set gene j's expression levels under the two conditions to $R_j\widetilde{\mathbf{X}}_j^1$ and $\widetilde{\mathbf{X}}_j^2$ if $U_j = 0$ or $\widetilde{\mathbf{X}}_j^1$ and $R_j\widetilde{\mathbf{X}}_j^2$ if $U_j = 1$. If gene j is not a true DEG, its expression levels under the two conditions would remain unchanged, i.e., $\widetilde{\mathbf{X}}_j^1$ and $\widetilde{\mathbf{X}}_j^2$. Such data generation strategy has no guarantee of i.i.d. read counts for non-DEGs if the samples in $\mathbf{X}^2$ have batch effects.

(iv) We repeated Step (iii) for 100 times to generate 100 semi-synthetic datasets.

The human monocyte RNA-seq dataset is available in the NCBI Sequence Read Archive (SRA) under accession number SRP082682 (https://www.ncbi.nlm.nih.gov/Traces/study/?acc=srp082682). The yeast RNA-seq data is available in the European Nucleotide Archive (ENA) archive with project ID PRJEB5348 (https://www.ebi.ac.uk/ena/browser/view/PRJEB5348).

Single-cell RNA-seq data with synthetic spike-in DEGs

We used scDesign2, a flexible probabilistic simulator to generate realistic scRNA-seq count data with gene correlations captured [19]. Using scDesign2, we generated two sets of semi-synthetic data from two peripheral blood mononuclear cell (PBMC) real datasets [20]: one generated using the 10x Genomics protocol [21] and the other using Drop-seq [22]. Each synthetic dataset contains two types of cells: CD4+ T cells, and cytotoxic T cells, which we treated as two conditions. Starting with the real data generated using either 10x Genomics or Drop-seq, we used the following steps to generate synthetic scRNA-seq data.

(i) First, we fit the real data count matrices using `R` function `fit_model_scDesign2` for each cell type by specifying the underlying distribution of each gene as negative binomial. Denote the resulting marginal distributions of gene j as $NB(\hat{\mu}_{j1}, \hat{\theta}_{j1})$ for CD4+ T cells and $NB(\hat{\mu}_{j2}, \hat{\theta}_{j2})$ for cytotoxic T cells, $j = 1, \ldots, d$. The gene-gene correlations with each cell type were fitted using a copula model.

(ii) Let $\mathbf{X}^{\text{cd4}}$ and $\mathbf{X}^{\text{cyto}}$ denote the read count matrices of CD4+ T cells and cytotoxic T cells. To define true DEGs, we first computed the log fold change of gene j by $\text{logFC}_j = \log_2 \left[(\bar{\mathbf{X}}_j^{\text{cd4}} + 1)/(\bar{\mathbf{X}}_j^{\text{cyto}} + 1) \right]$ for $j = 1, \ldots, d$, where $\mathbf{X}_j^{\text{cd4}}$ and $\mathbf{X}_j^{\text{cyto}}$ denote the j-th row vector of $\mathbf{X}^{\text{cd4}}$ and $\mathbf{X}^{\text{cyto}}$ respectively and $\bar{\cdot}$ denotes the average of elements in a vector. We then selected 1000 genes with the largest absolute fold changes as true DEGs and kept the remaining ones as true non-DEGs.

(iii) We simulated the semi-synthetic datasets using `R` function `simulate_count_scDesign2`.

Specifically, we set the number of synthetic cells generated by scDesign2 equal to the number of real cells for each cell type. If a gene j is a true DEG, we specify its marginal distributions under the two conditions as $NB(\hat{\mu}_{j1}, \hat{\theta}_{j1})$ and $NB(\hat{\mu}_{j2}, \hat{\theta}_{j2})$ respectively. If a gene j is a true non-DEG, we specify its marginal distribution under both conditions as $NB((\hat{\mu}_{j1} + \hat{\mu}_{j2})/2, (\hat{\theta}_{j1} + \hat{\theta}_{j2})/2)$. We used the fitted copula models from the two cell types to generate genes' (correlated) expression read counts.

(iv) We repeated Step (iii) for 200 times to generate 200 semi-synthetic datasets.

Both `fit_model_scDesign2` and `simulate_count_scDesign2` come from `R` package `scDesign2` [19]. The 10x Genomic PBMC dataset and the Drop-seq PBMC dataset are available from the Gene Expression Omnibus (GEO) with accession number GSE132044 (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE132044) and the Single Cell Portal with accession numbers SCP424 (https://singlecell.broadinstitute.org/single_cell/study/SCP424/single-cell-comparison-pbmc-data).

S2.5.6 Implementation of Clipper in omics data applications

Below we briefly introduce the implementation of Clipper in the four omics data applications. All the results were obtained by running using `R` package `Clipper` (see package vignette for details: https://github.com/JSB-UCLA/Clipper/blob/master/vignettes/Clipper.pdf).

Peak calling from ChIP-seq data

(i) We consider each genomic location, i.e., a base pair, as a feature and each ChIP-seq sample as a replicate under the experimental or background condition. Then we consider the read count of each location in each sample as the corresponding feature's measurement. Doing so, we summarized ChIP-seq data into a $d \times (m + n)$ matrix, where d is the number of locations, and m and n are the numbers of experimental and control samples, respectively. We then applied Clipper to perform an enrichment analysis to obtain the contrast score C_j for each location j. In our study, $m = n = 1$, so the default Clipper implementation is Clipper-minus-BC.

(ii) For any target FDR threshold q, Clipper gives a cutoff T_q on contrast scores.

(iii) We then used existing peak calling methods, e.g., MACS2 and HOMER, to call candidate peaks with the least stringent q-value cutoff. For example, when we used MACS2, we set the q-value cutoff as 1.

(iv) We computed the contrast score of each candidate peak as the median of the contrast scores of all the locations within.

(v) The candidate peaks with contrast scores greater than or equal to T_q are called discoveries.

DEG identification from bulk RNA-seq data

(i) We consider each gene as a feature and the class label—classical and non-classical human monocytes—as the two conditions. Then we consider $\log_2$-transformed read counts with a pseudocount 1 as measurements. Doing so, we summarized the gene expression matrix into a $d \times (m + n)$ matrix, where d is the number of genes, and m and n are the numbers of samples under the two conditions, respectively. We then applied Clipper to perform a differential analysis to obtain a contrast score C_j for each gene. In our study, $m = n = 3$, so the default Clipper implementation is Clipper-max-GZ with $h = 1$.

(ii) For any target FDR threshold q, Clipper gives a cutoff T_q on contrast scores.

(iii) The genes with contrast scores greater than or equal to T_q are called discoveries.

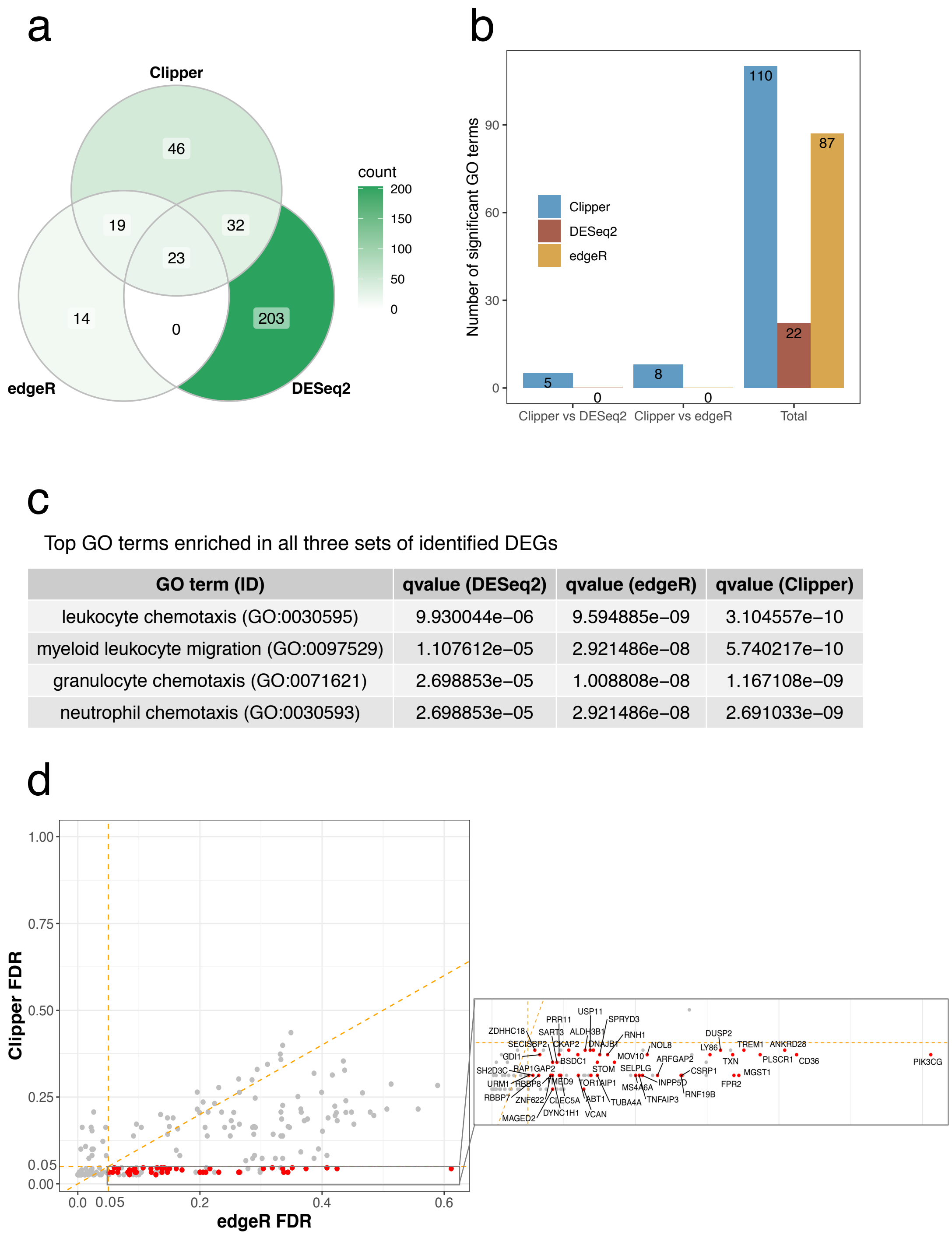
DEG identification from scRNA-seq data

(i) We consider each gene as a feature and the cell type—CD4+ T cells and cytotoxic T cells—as the two conditions. We first performed the TMM normalization [18]. Then we consider $\log_2$-transformed read counts with a pseudocount 1 as measurements. Doing so, we summarized the gene expression matrix into a $d \times (m+n)$ matrix, where d is the number of genes, and m and n are the numbers of samples under the two conditions, respectively. We then applied Clipper to perform differential analysis to obtain a contrast score C_j for each gene j. In our study, $m = 1172$, $n = 789$ for Drop-seq dataset and $m = 963$, $n = 694$ for 10x Genomics dataset. The default Clipper implementation is Clipper-max-GZ with $h = 1$, the default number of permutations.

(ii) For any target FDR threshold q, Clipper gives a cutoff T_q on contrast scores.

(iii) The genes with contrast scores greater than or equal to T_q are called discoveries.

S2.5.7 Supplementary figures



Top GO terms enriched in all three sets of identified DEGs

GO term (ID)	qvalue (DESeq2)	qvalue (edgeR)	qvalue (Clipper)
leukocyte chemotaxis (GO:0030595)	9.930044e−06	9.594885e−09	3.104557e−10
myeloid leukocyte migration (GO:0097529)	1.107612e−05	2.921486e−08	5.740217e−10
granulocyte chemotaxis (GO:0071621)	2.698853e−05	1.008808e−08	1.167108e−09
neutrophil chemotaxis (GO:0030593)	2.698853e−05	2.921486e−08	2.691033e−09

Figure 2.5: Application of Clipper, DESeq2, and edgeR to identifying DEGs from the classical and non-classical human monocyte dataset.

(a) A Venn diagram showing the overlaps of the identified DEGs (at the FDR threshold $q = 5\%$) by the three DE methods. **(b)** Numbers of GO terms enriched (with enrichment q-values < 0.01) in the DEGs found by Clipper, DESeq2 and edgeR (column 3), or in the DEGs specifically identified by Clipper or DESeq2/edgeR in the pairwise comparison between Clipper and DESeq2 (column 1) or between Clipper and edgeR (column 2). More GO terms are enriched in the DEGs identified by Clipper than in those identified by edgeR or DESeq2. **(c)** Enrichment q-values of four GO terms that are found enriched (with enrichment q-values < 0.01) in all three sets of identified DEGs, one set per method. All the four terms are most enriched in the DEGs identified by Clipper. **(d)** A scatterplot of the claimed FDR of Clipper against that of edgeR for all the DEGs identified by Clipper, edgeR or DESeq2. The 46 DEGs only identified by Clipper are highlighted with red.

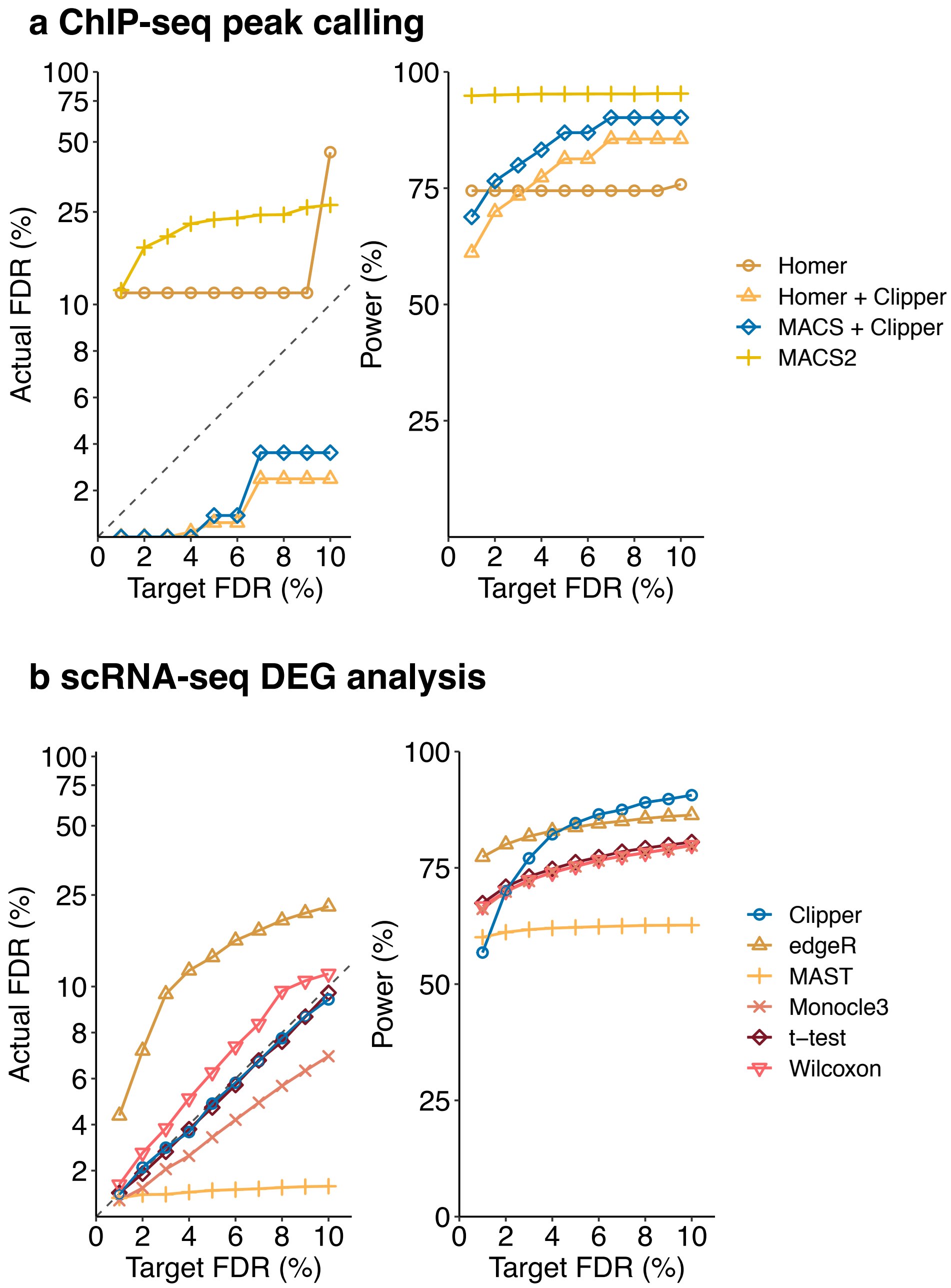


Figure 2.6: Comparison of Clipper and popular bioinformatics methods in terms of FDR control and power.

(a) peaking calling analysis on semi-synthetic ChIP-seq data; **(b)** DEG analysis on synthetic 10x Genomics scRNA-seq data; In all four panels, the target FDR threshold q ranges from 1% to 10%. In the "Actual FDR vs. Target FDR" plot of each panel, points above the dashed diagonal line indicate failed FDR control; when this happens, the power of the corresponding methods is not shown, including HOMER in (a), MACS2 for target FDR less than 5% in (a), edgeR in (c), and multiHICcompare, and FIND in (d). In all four applications, Clipper controls the FDR while maintaining high power, demonstrating Clipper's broad applicability in high-throughput data analyses

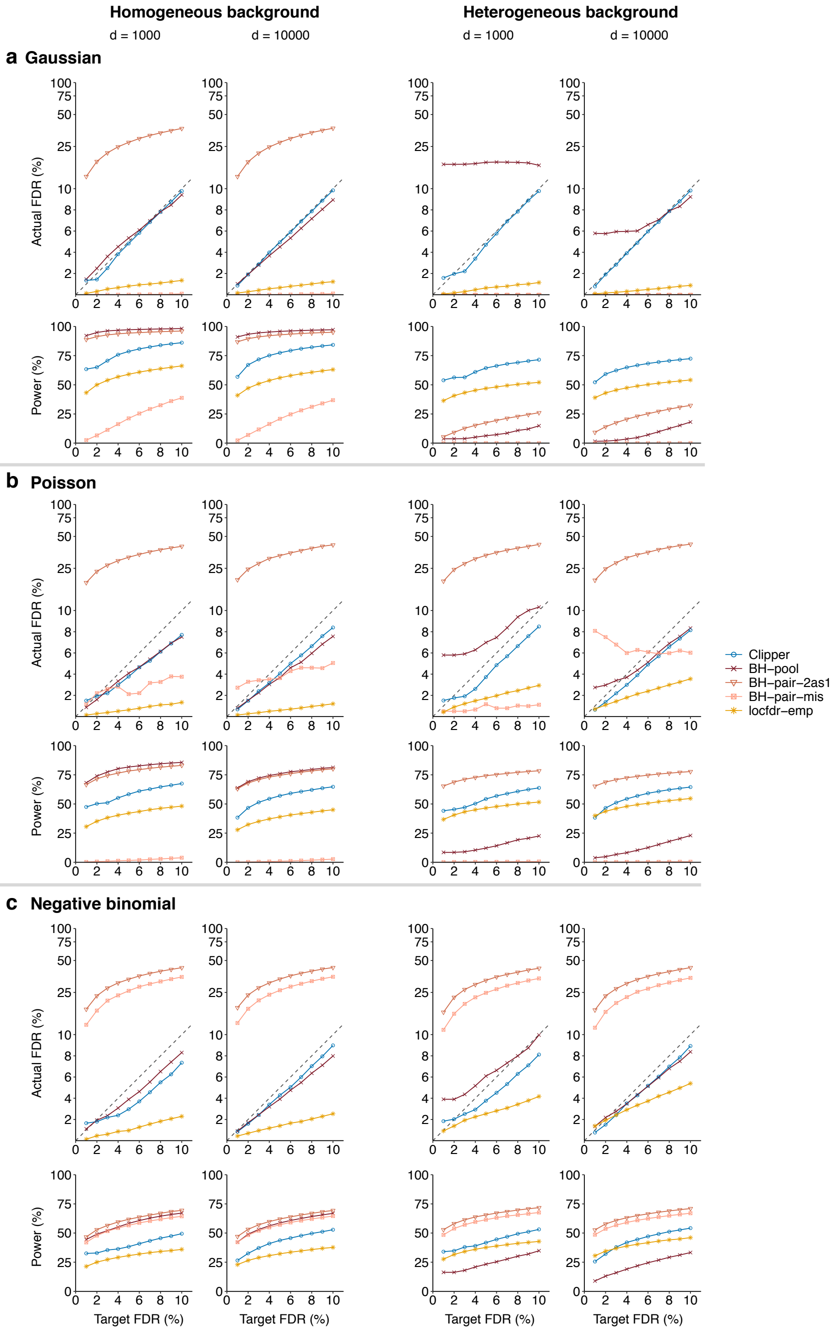


Figure 2.7: In the 1vs1 enrichment analysis, comparison of Clipper and four other generic FDR control methods (BH-pool, BH-pair-2as1, BH-pair-mis, and locfdr-emp) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 1000$ or 10,000 features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

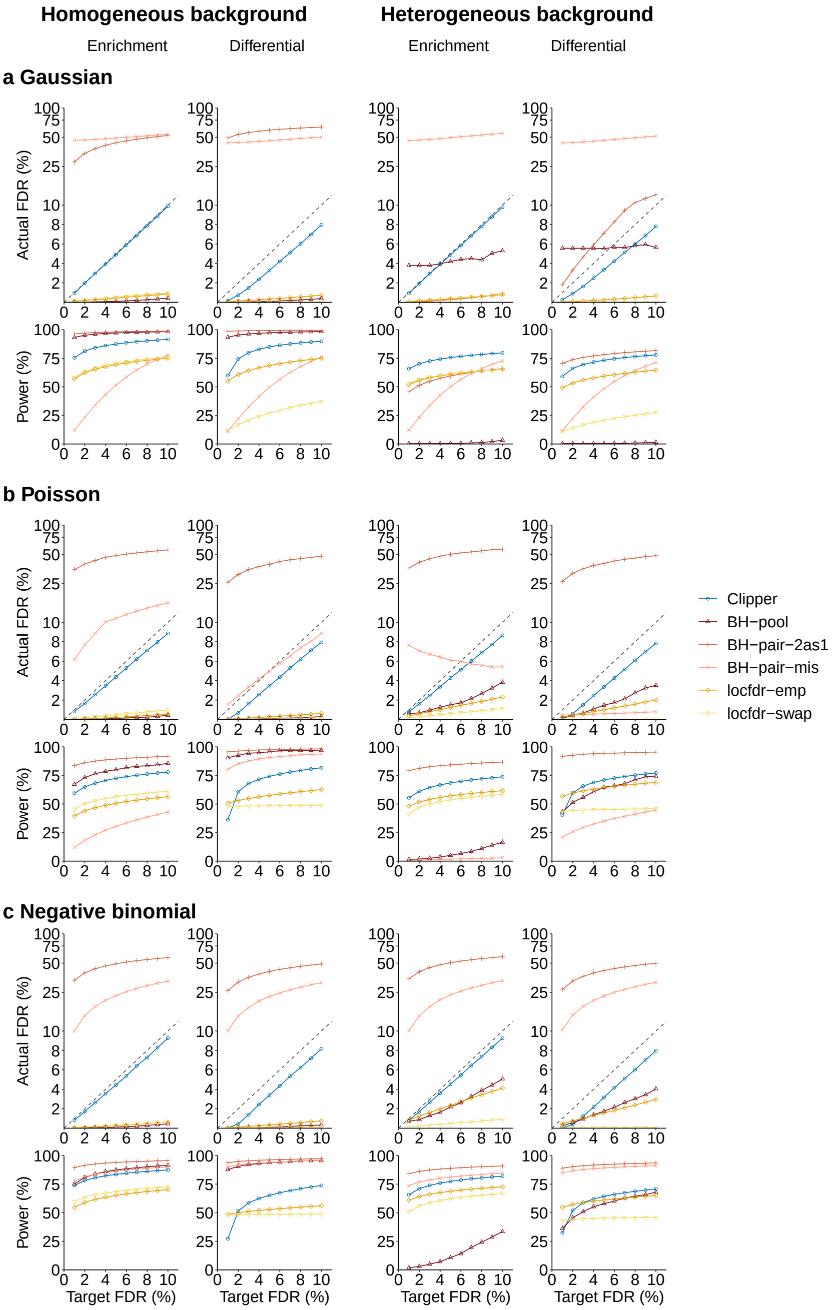


Figure 2.8: In the 2vs1 enrichment analysis (columns 1 and 3) and differential analysis (columns 2 and 4), comparison of Clipper and five other generic FDR control methods (BH-pooled, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous(two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background (except for differential analysis with $q \leqslant 2\%$).

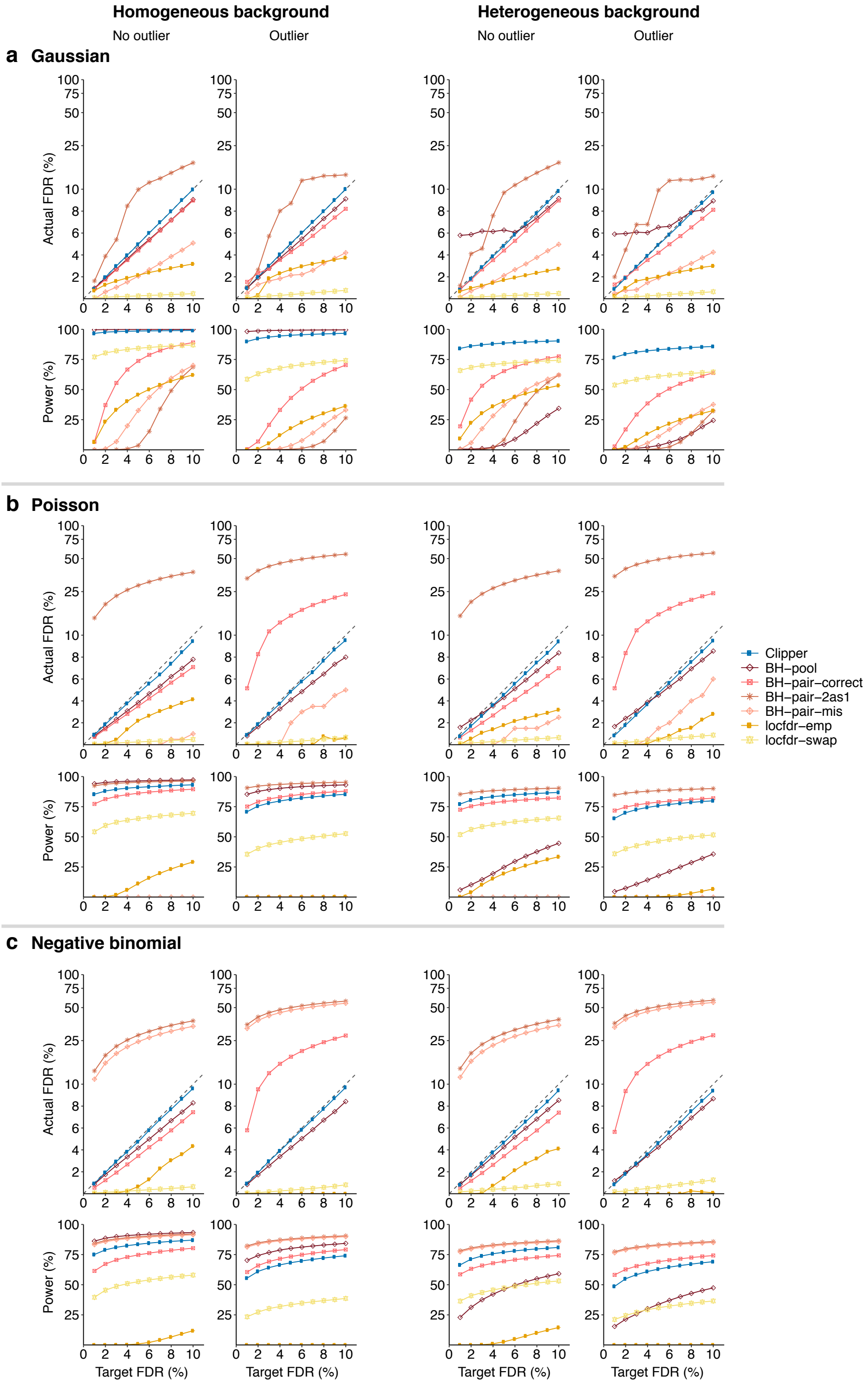


Figure 2.9: In the 3vs3 enrichment analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of Clipper and six other generic FDR control methods (BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap)in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. 10% of the features are interesting features. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

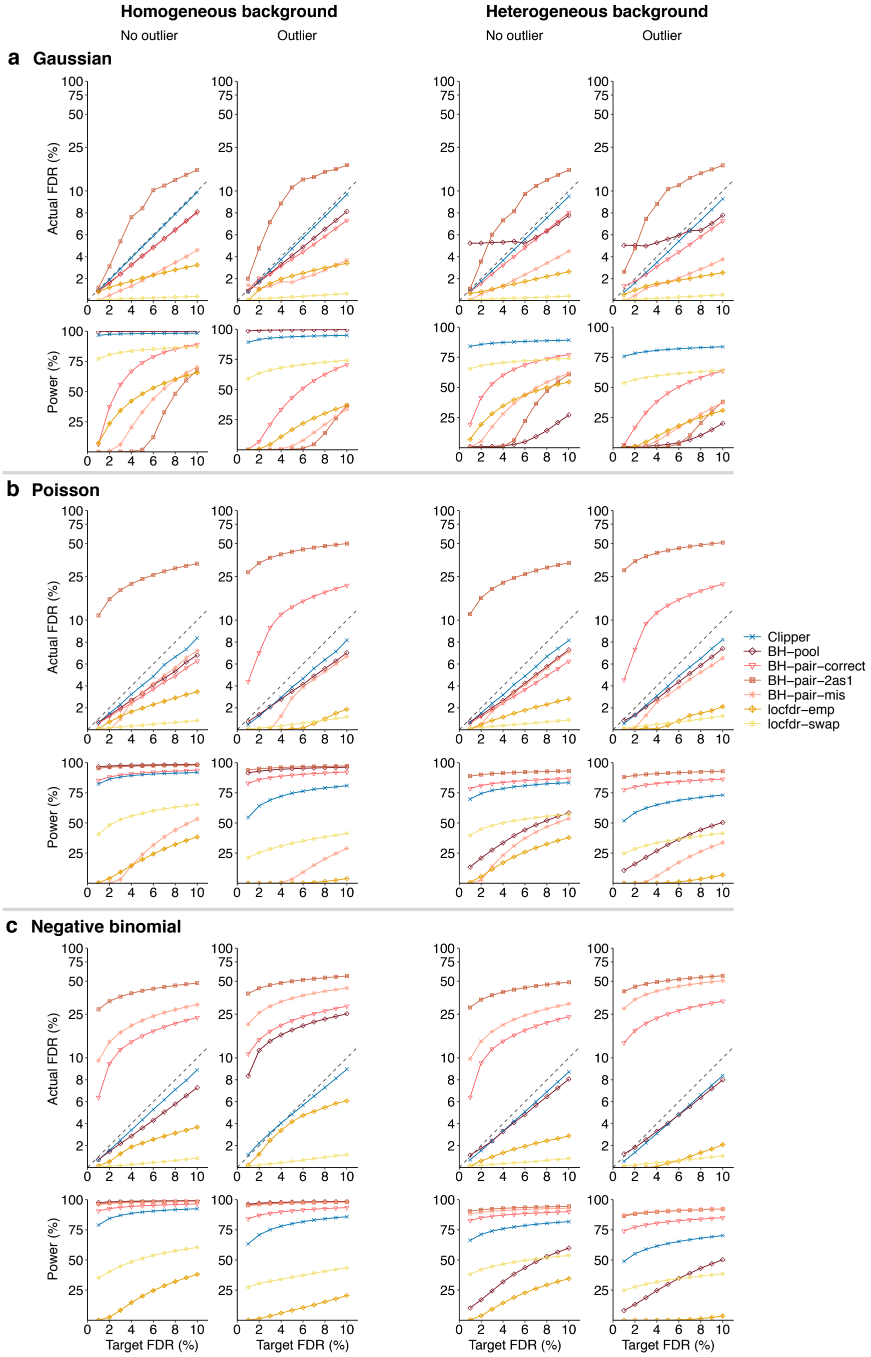


Figure 2.10: In the 3vs3 differential analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of Clipper and six other generic FDR control methods (BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background (except for Poisson distribution where Clipper is second to BH-pair-correct, an idealistic method).

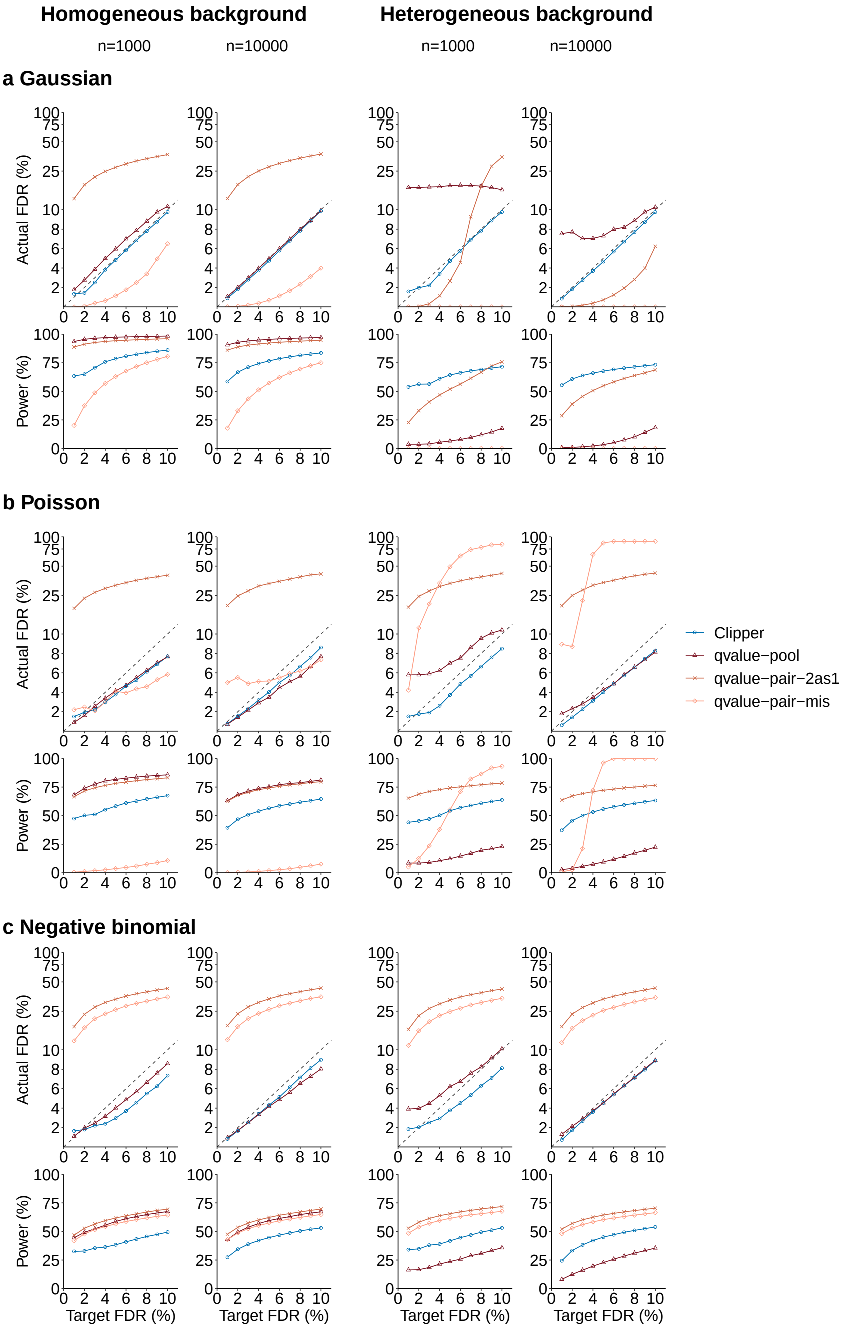


Figure 2.11: In the 1vs1 enrichment analysis, comparison of Clipper and three other generic FDR control methods using Storey's q-value (qvalue-pool, qvalie-pair-2as1, and qvalue-pair-mis) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 1000$ or 10,000 features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

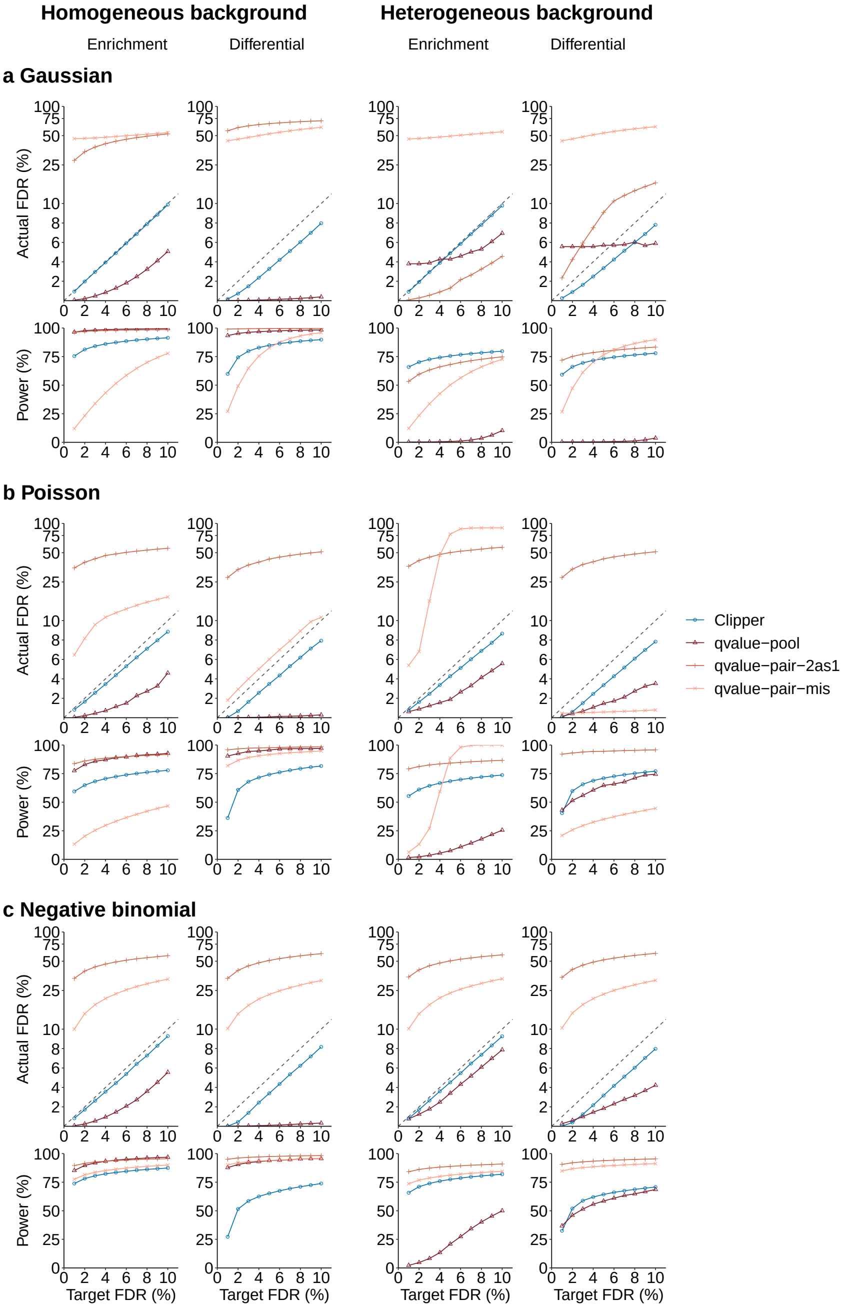


Figure 2.12: In the 2vs1 enrichment analysis (columns 1 and 3) and differential analysis (columns 2 and 4), comparison of Clipper and three other generic FDR control methods using Storey's q-value (qvalue-pool, qvalie-pair-2as1, and qvalue-pair-mis) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

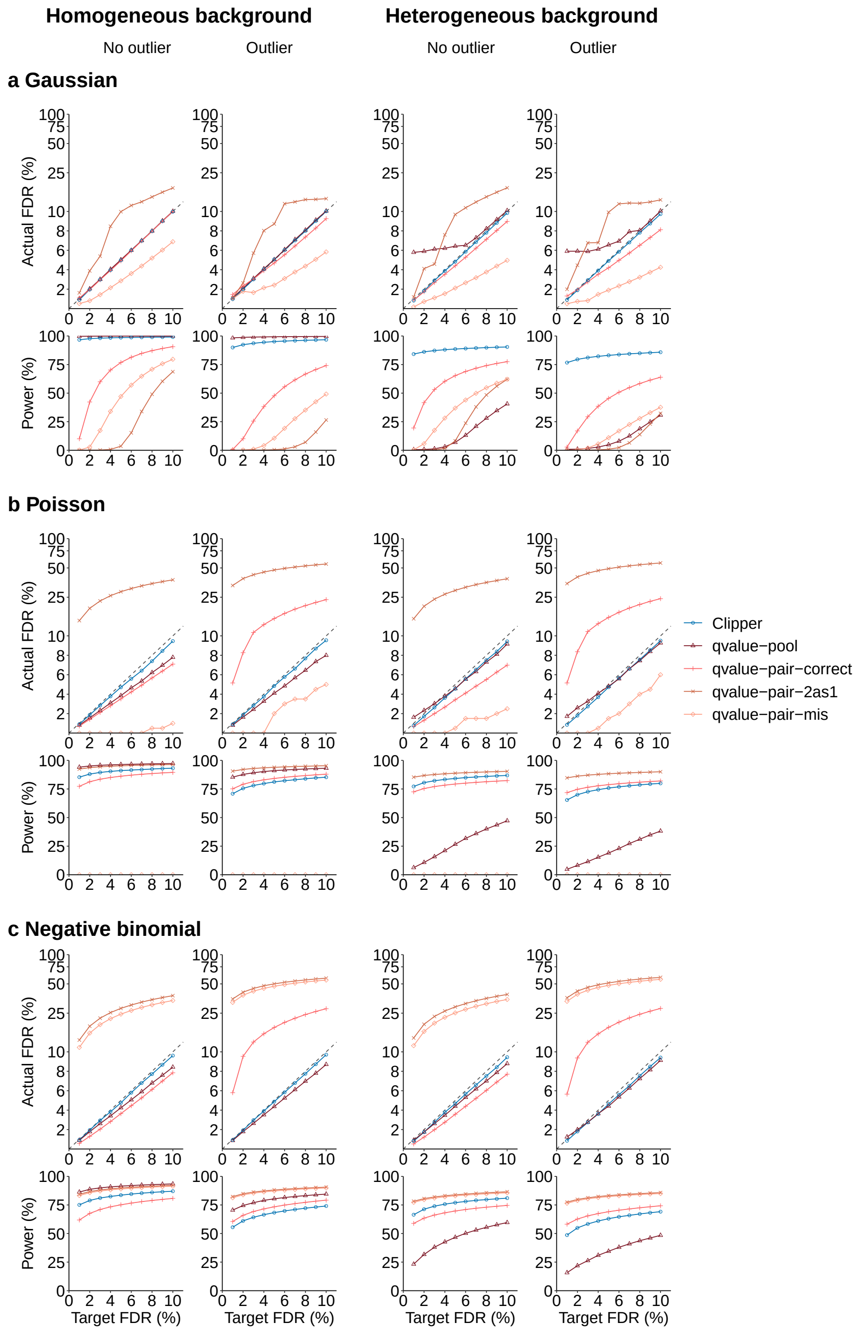


Figure 2.13: In the 3vs3 enrichment analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of Clipper and four other generic FDR control methods using Storey's q-value (qvalue-pooled, qvalue-pair-correct, qvalue-pair-2as1, and qvalue-pair-mis) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

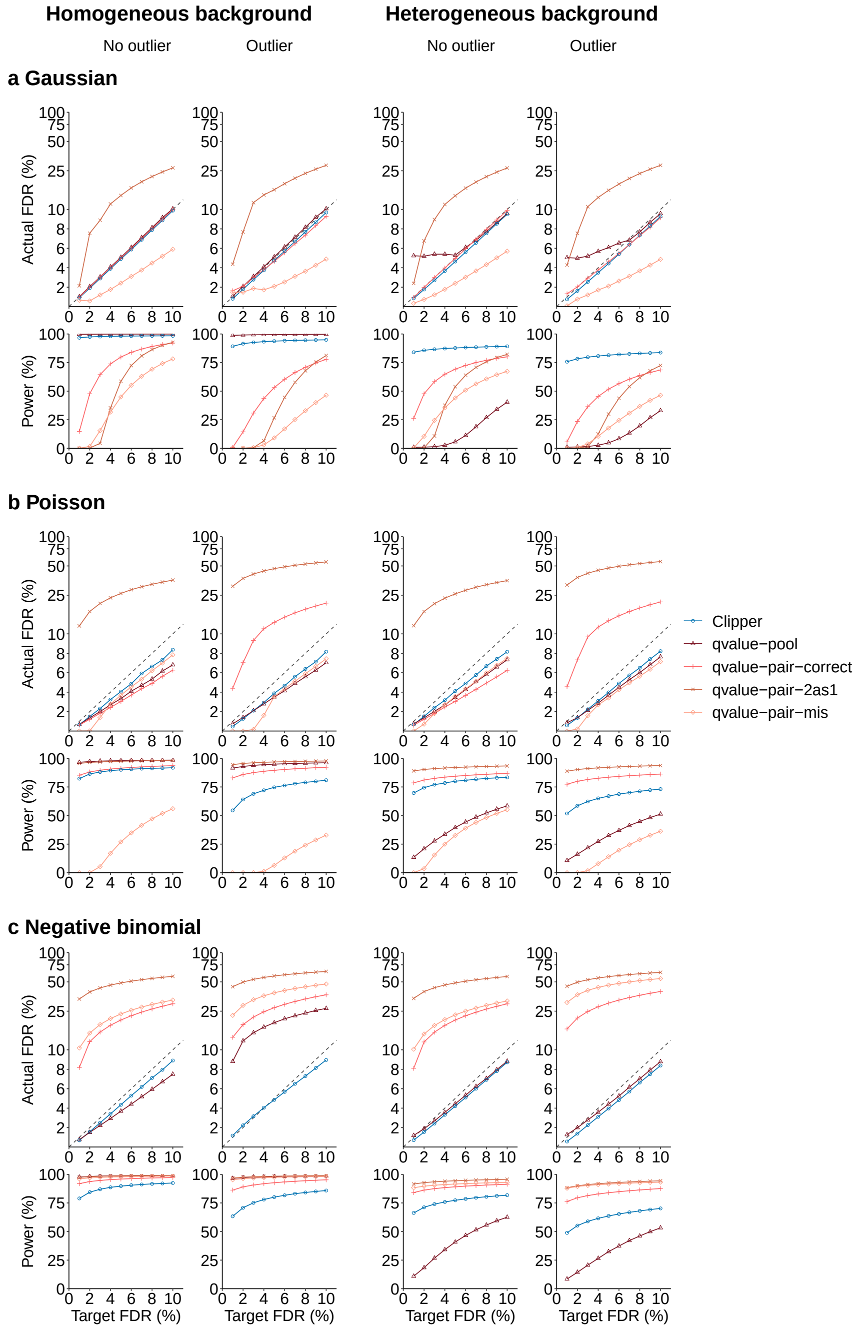


Figure 2.14: In the 3vs3 differential analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of Clipper and four other generic FDR control methods using Storey's q-value (qvalue-pooled, qvalue-pair-correct, qvalue-pair-2as1, and qvalue-pair-mis) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background (except for Poisson distribution where Clipper is second to qvalue-pair-correct, an idealistic method).

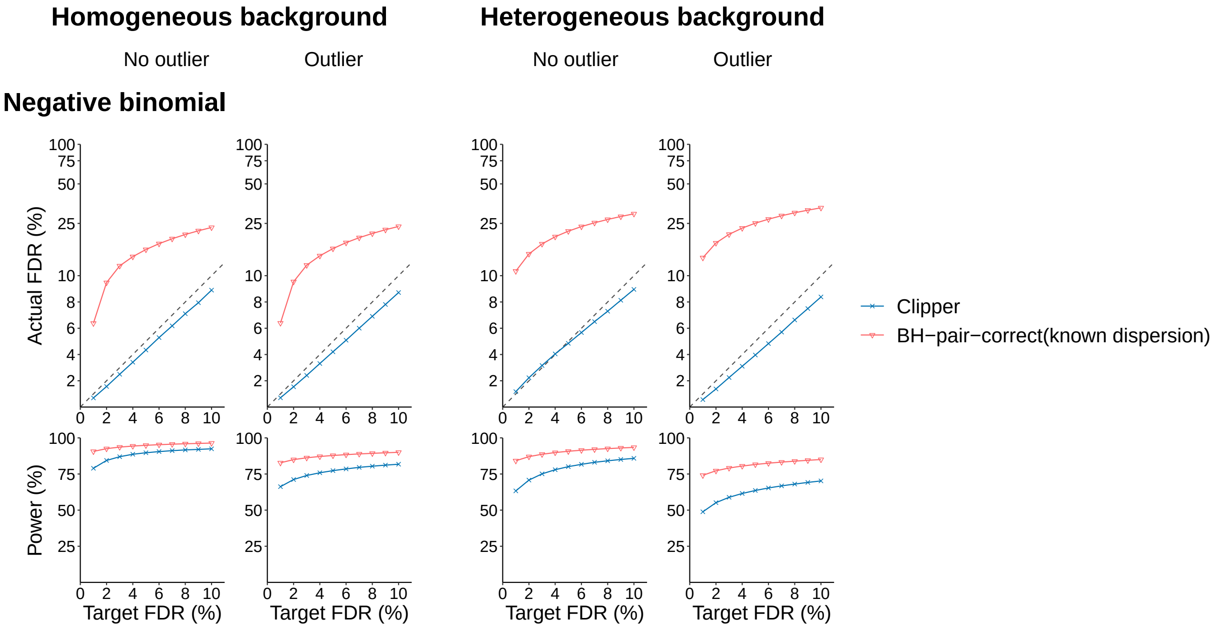


Figure 2.15: In the 3vs3 differential analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of Clipper, BH-pair-correct (known dispersion), and BH-pair-correct (unknown dispersion) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. BH-pair-correct (unknown dispersion) cannot control the FDR in all settings. In contrast, Clipper is consistently the most powerful for homogeneous and heterogeneous background.

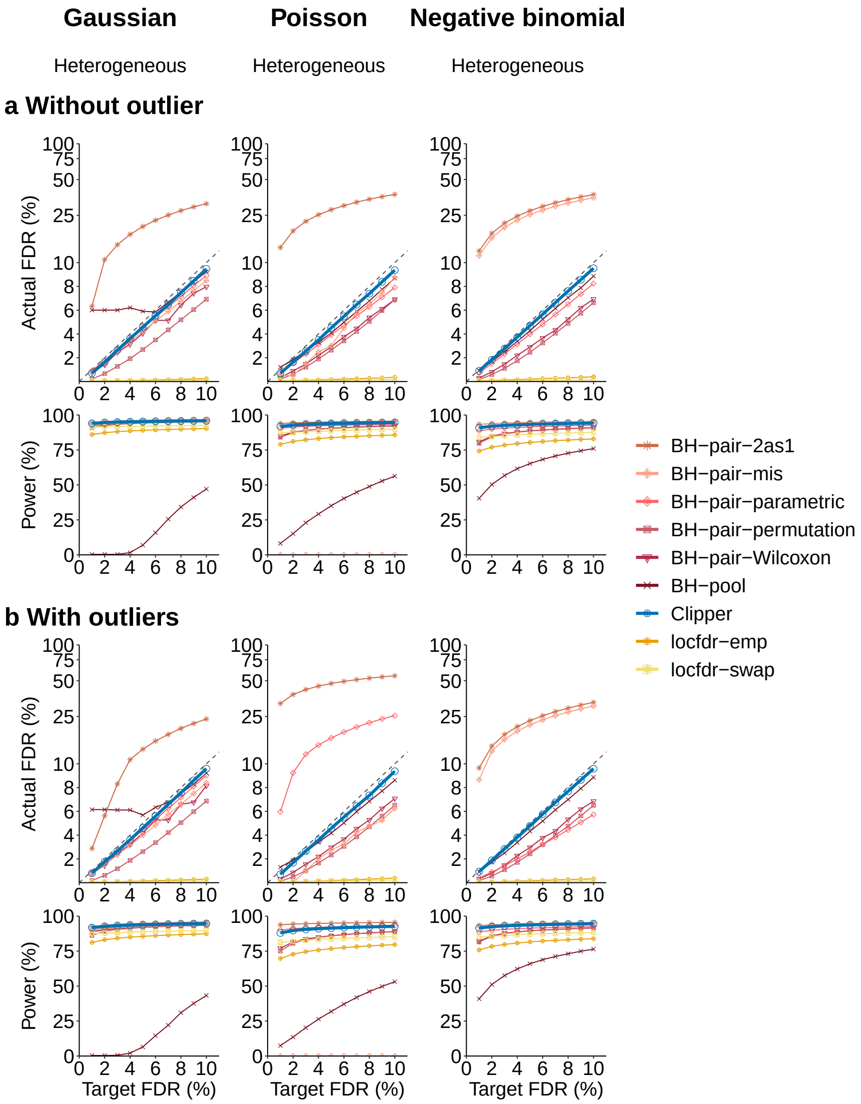


Figure 2.16: In the 10vs10 enrichment analysis with and without outliers, comparison of Clipper and eight generic FDR control methods (BH-pooled, BH-pair-Wilcoxon, BH-pair-parametric, and BH-pair-permutation, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from the Gaussian distribution (left), the Poisson distribution (middle), or the negative binomial distribution (right) under heterogeneous background scenarios. Clipper achieves the highest power for all three distributions.

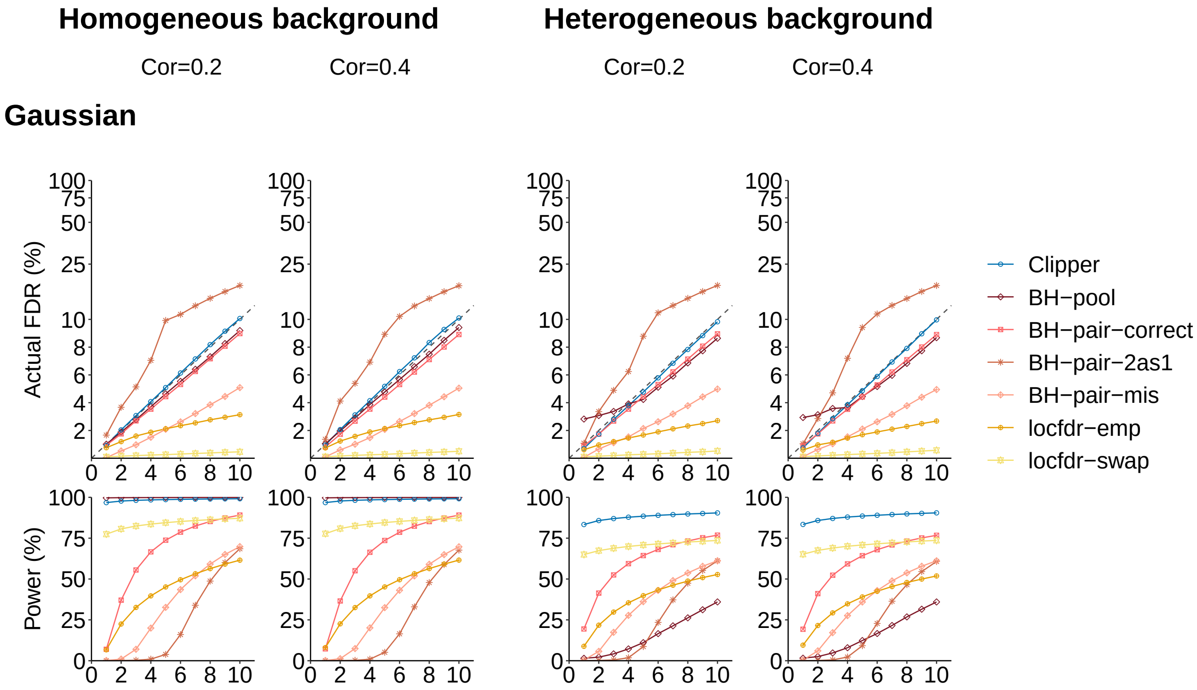


Figure 2.17: In the 3vs3 enrichment analysis with correlated features, comparison of Clipper and six other generic FDR control methods (BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power in 3vs3 enrichment analysis.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from a multivariate Gaussian distribution with a correlation 0.2 (columns 1 and 3) or 0.4 (columns 2 and 4) between features. Among the methods that control the FDR, Clipper is the second most powerful for homogeneous background and the most powerful for heterogeneous background.

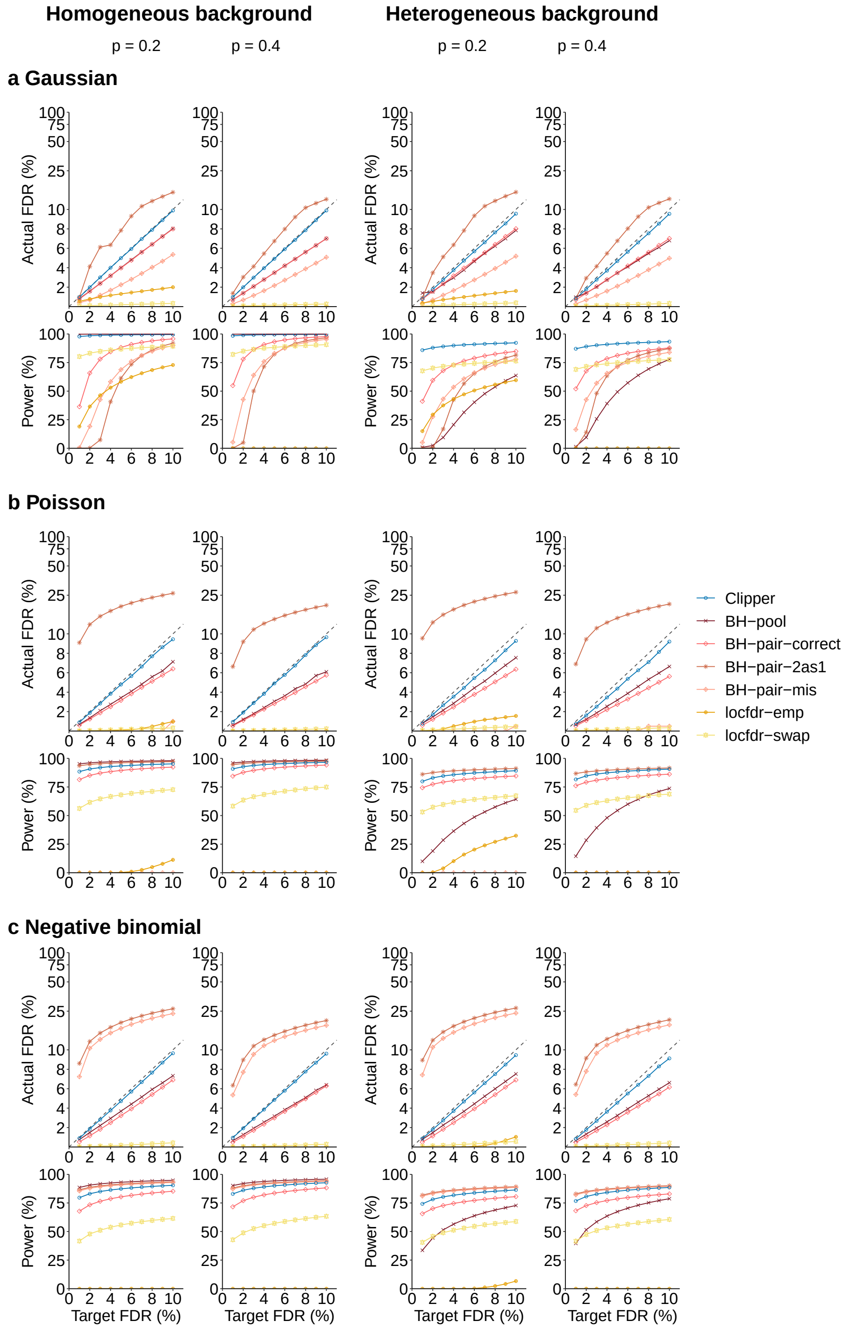


Figure 2.18: In 3vs3 enrichment analysis with different proportions of interesting features without outliers, comparison of Clipper and six generic FDR control methods (BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from the Gaussian distribution, the Poisson distribution, or the negative binomial distribution, with the proportion of interesting features being 0.2 (columns 1 and 3) or 0.4 (columns 2 and 4) under homogeneous (columns 1 and 2) and heterogeneous (columns 3 and 4) background scenarios. Clipper achieves the highest power for all distributions.

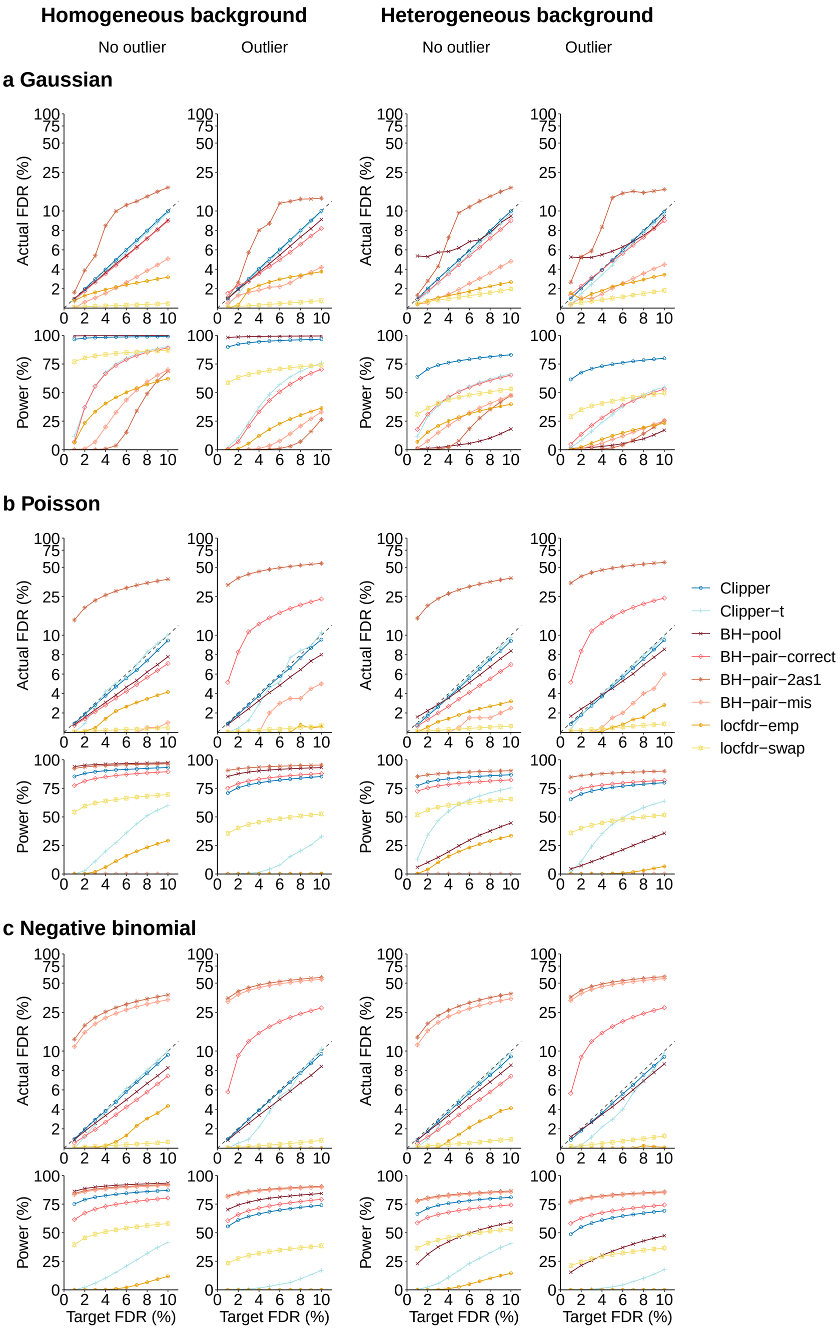


Figure 2.19: In the 3vs3 enrichment analysis with and without outliers, comparison of the default Clipper, the Clipper variant using the t statistic as the contrast score (Clipper-t), and six generic FDR control methods (Clipper BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from the Gaussian distribution, the Poisson distribution, or the negative binomial distribution under homogeneous (columns 1 and 2) and heterogeneous (columns 3 and 4) background scenarios. Clipper achieves higher power than Clipper-t does.

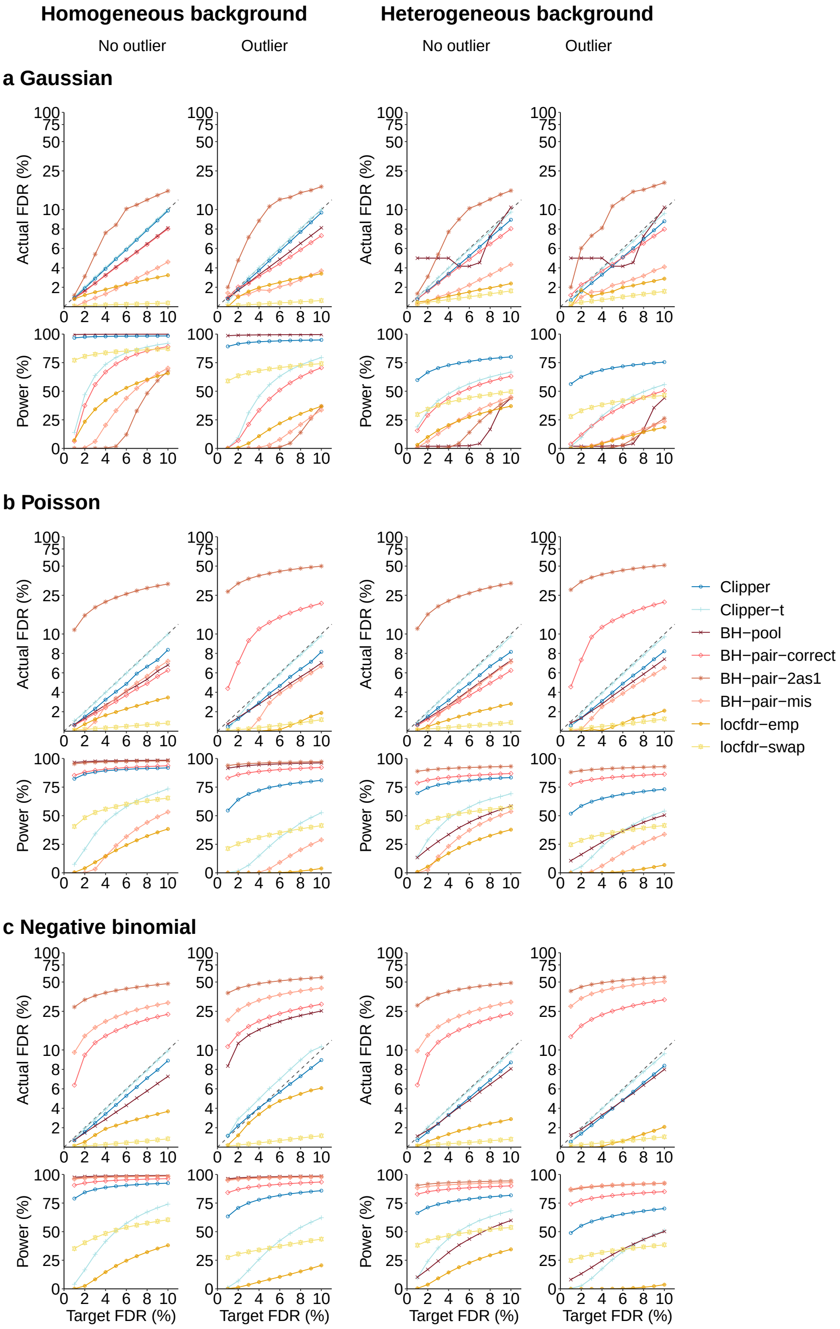


Figure 2.20: In the 3vs3 differential analysis with and without outliers, comparison of the default Clipper, the Clipper variant using the t statistic to calculate the degree of interestingness (Clipper-t), and six generic FDR control methods (Clipper BH-pooled, BH-pair-correct, BH-pair-2as1, BH-pair-mis, locfdr-emp, and locfdr-swap) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from the Gaussian distribution, the Poisson distribution, or the negative binomial distribution under homogeneous (columns 1 and 2) and heterogeneous (columns 3 and 4) background scenarios. Clipper achieves higher power than Clipper-t does.

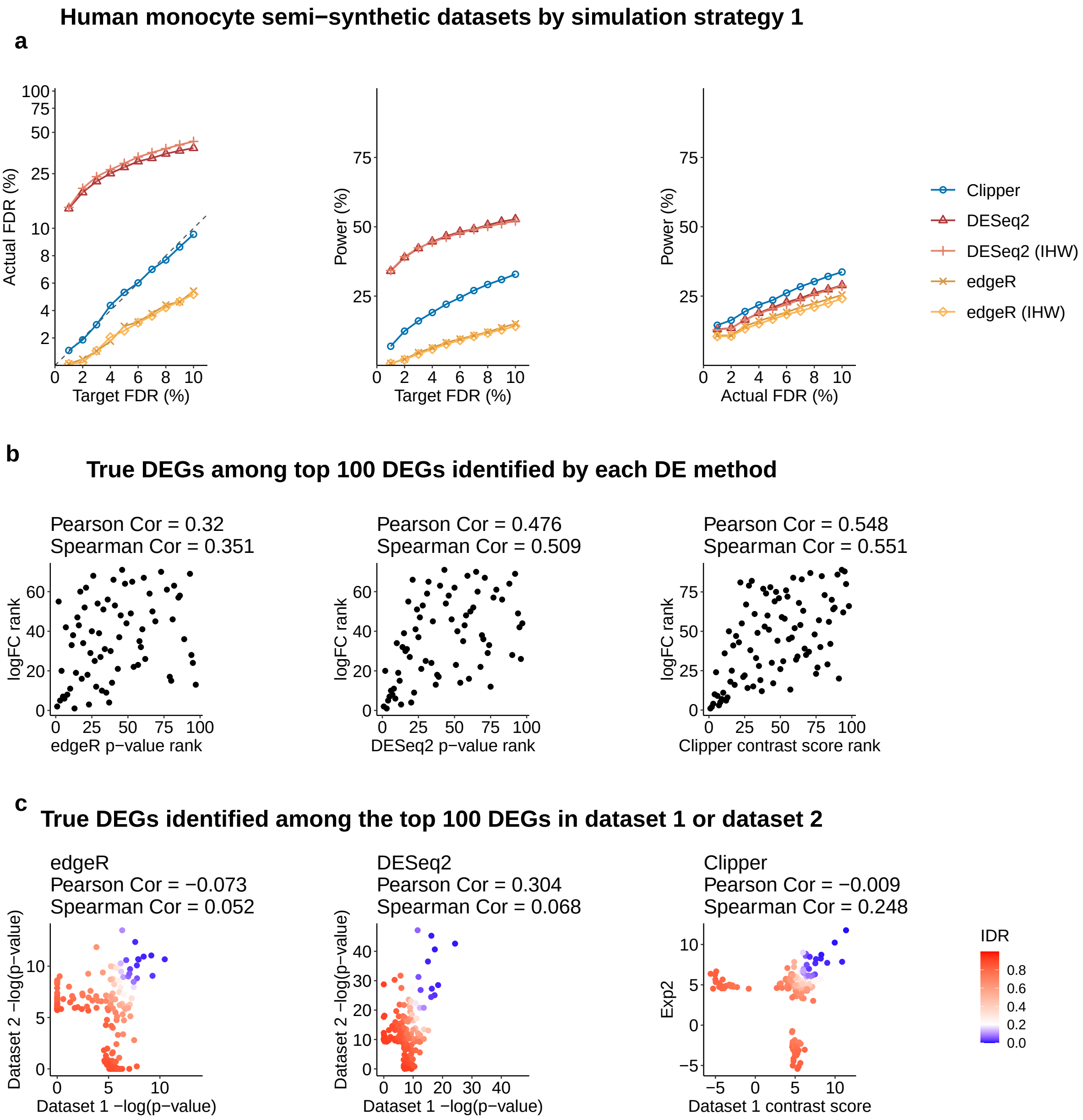


Figure 2.21: Comparison of Clipper and two popular DEG identification methods—edgeR and DESeq2—in DEG analysis on semi-synthetic bulk RNA-seq data (generated from human monocyte real data using simulation strategy 1.

(a) FDR control, power given the same target FDR, and power given the same actual FDR. **(b)** Ranking consistency of the true DEGs among the top 100 DEGs identified by each method. The consistency is defined between the genes' ranking based on edgeR/DESeq2's p-values or Clipper's contrast scores and their ranking based on true expression fold changes. **(c)** Reproducibility between two semi-synthetic datasets as technical replicates. Three reproducibility criteria are used: the IDR, Pearson correlation, and Spearman correalation. Each criterion is calculated for edgeR/DESeq2's p-values or Clipper's contrast scores on the two semi-synthetic datasets. Among the three methods, only Clipper controls the FDR, and Clipper achieves the highest power, the best gene ranking consistency, and the best reproducibility.

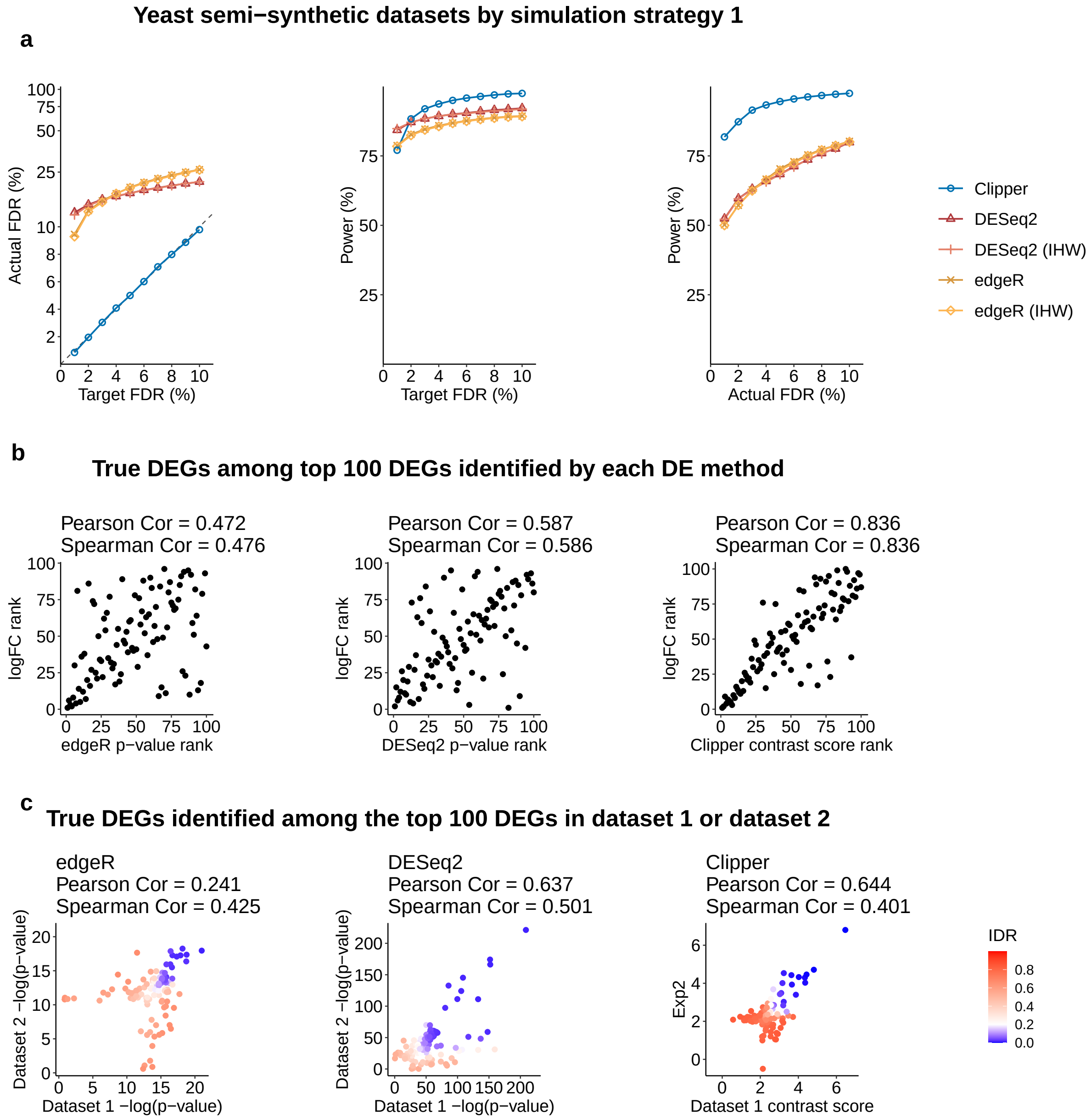


Figure 2.22: Comparison of Clipper and two popular DEG identification methods—edgeR and DESeq2—in DEG analysis on semi-synthetic bulk RNA-seq data (generated from yeast real data using simulation strategy 1.

(a) FDR control, power given the same target FDR, and power given the same actual FDR. **(b)** Ranking consistency of the true DEGs among the top 100 DEGs identified by each method. The consistency is defined between the genes' ranking based on edgeR/DESeq2's p-values or Clipper's contrast scores and their ranking based on true expression fold changes. **(c)** Reproducibility between two semi-synthetic datasets as technical replicates. Three reproducibility criteria are used: the IDR, Pearson correlation, and Spearman correalation. Each criterion is calculated for edgeR/DESeq2's p-values or Clipper's contrast scores on the two semi-synthetic datasets. Among the three methods, only Clipper controls the FDR, and Clipper achieves the highest power, the best gene ranking consistency, and the best reproducibility.

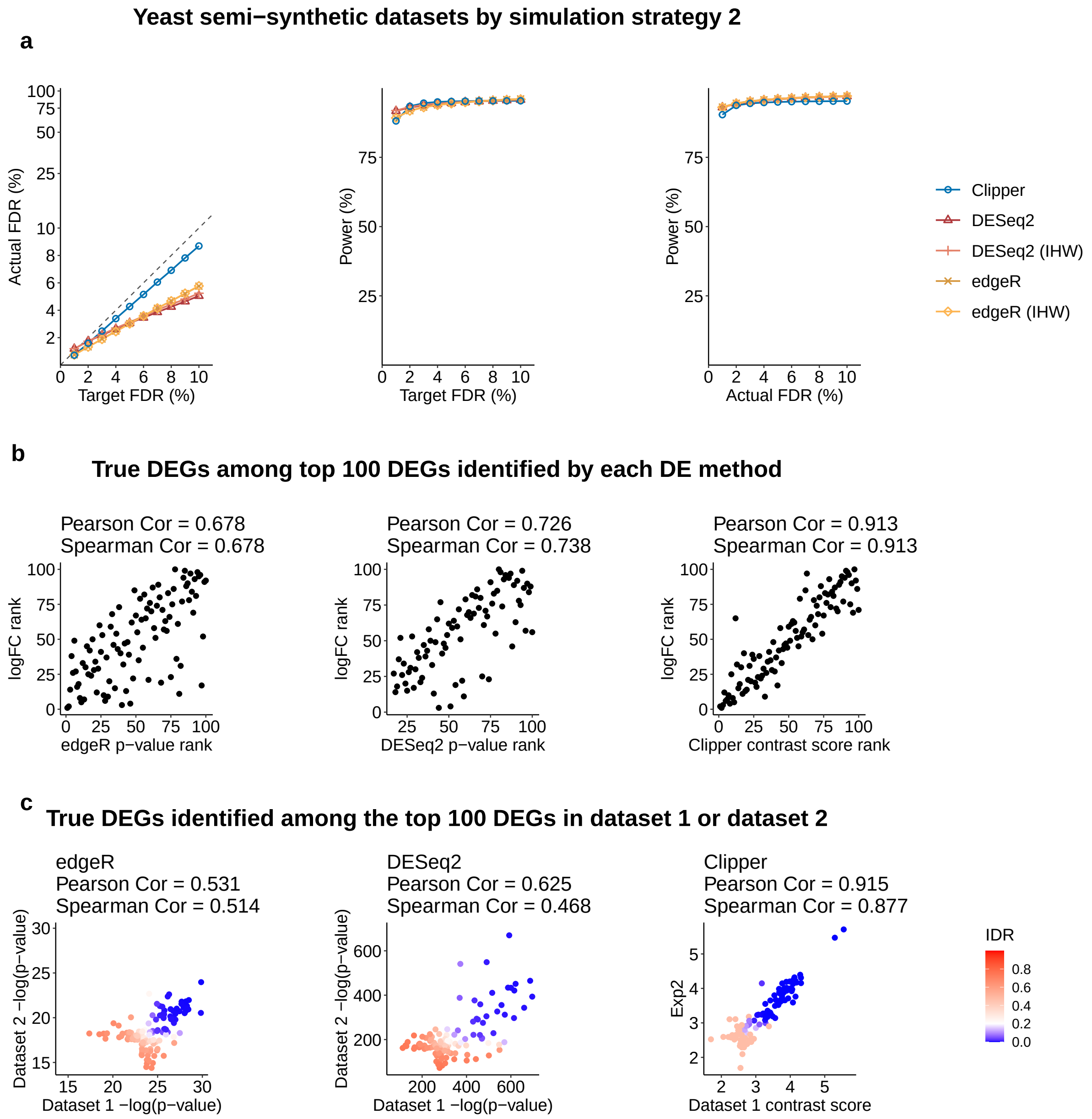


Figure 2.23: Comparison of Clipper and two popular DEG identification methods—edgeR and DESeq2—in DEG analysis on semi-synthetic bulk RNA-seq data (generated from yeast real data using simulation strategy 2.

(a) FDR control, power given the same target FDR, and power given the same actual FDR. **(b)** Ranking consistency of the true DEGs among the top 100 DEGs identified by each method. The consistency is defined between the genes' ranking based on edgeR/DESeq2's p-values or Clipper's contrast scores and their ranking based on true expression fold changes. **(c)** Reproducibility between two semi-synthetic datasets as technical replicates. Three reproducibility criteria are used: the IDR, Pearson correlation, and Spearman correalation. Each criterion is calculated for edgeR/DESeq2's p-values or Clipper's contrast scores on the two semi-synthetic datasets. Among the three methods, only Clipper controls the FDR, and Clipper achieves the highest power, the best gene ranking consistency, and the best reproducibility.

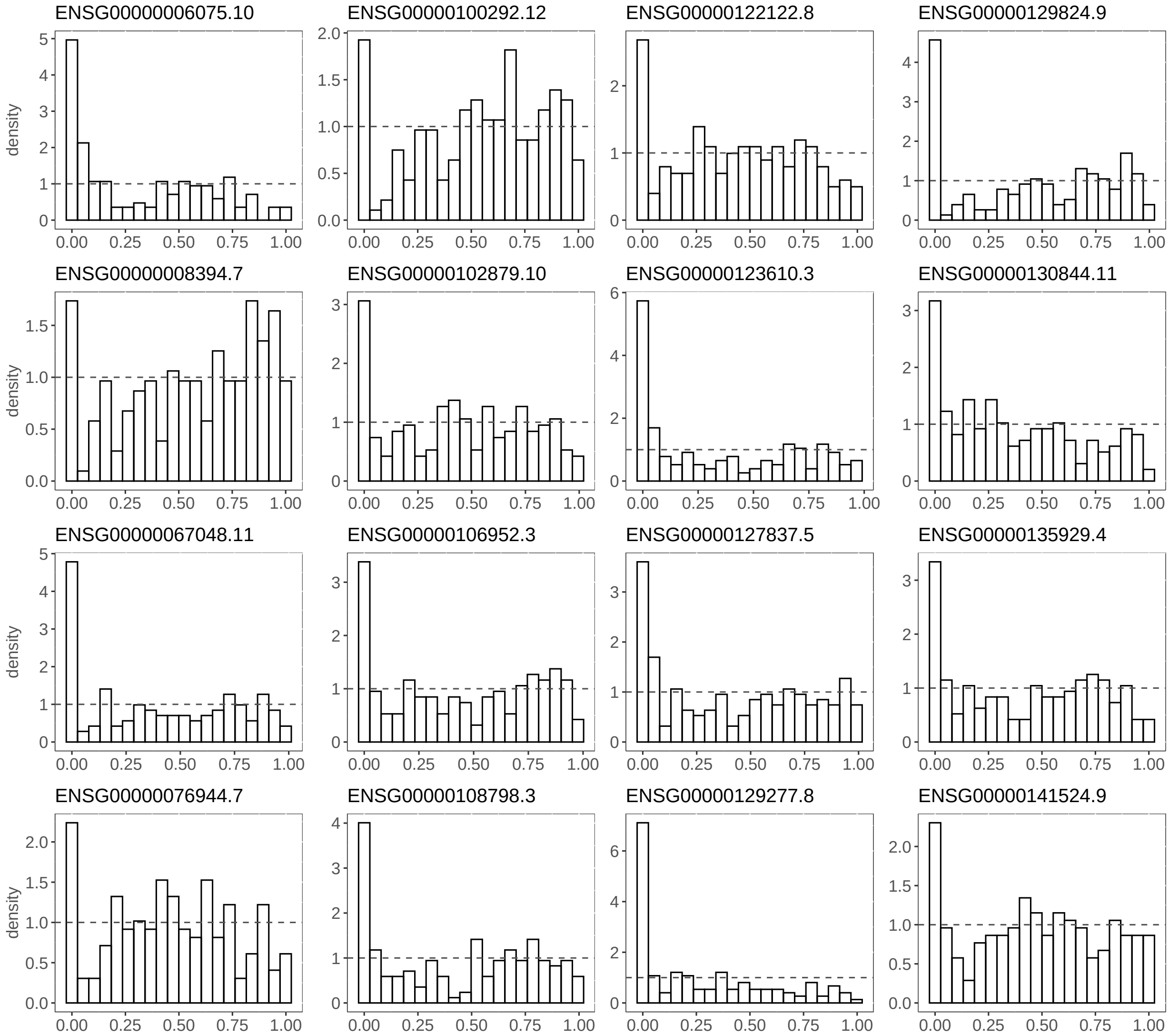


Figure 2.24: The p-value distributions of 16 non-DEGs that are most frequently identified by DESeq2 at $q = 5\%$ from 200 semi-synthetic datasets. The p-values of these 16 genes tend to be overly small, and their distributions are non-uniform with a mode close to 0.

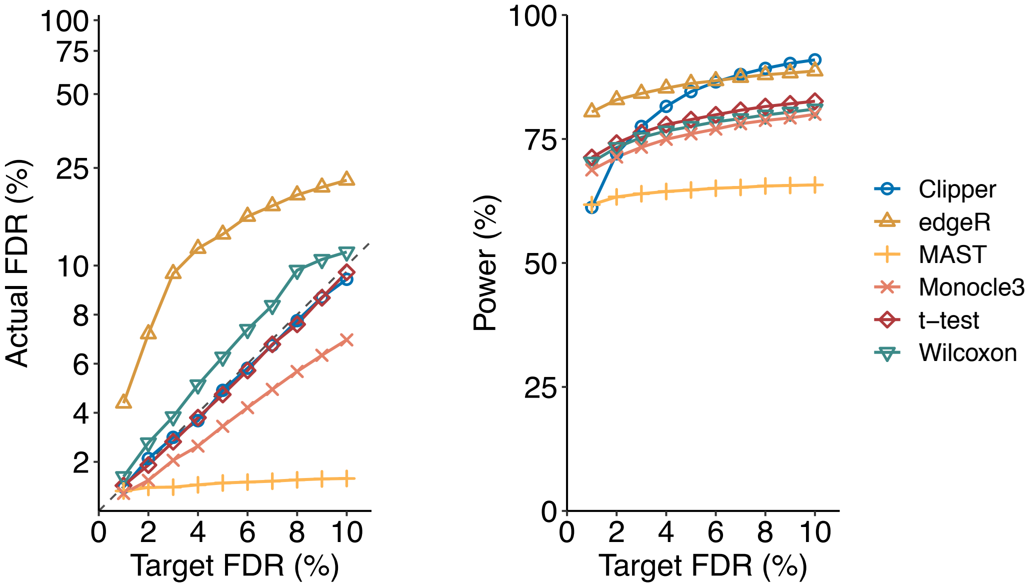
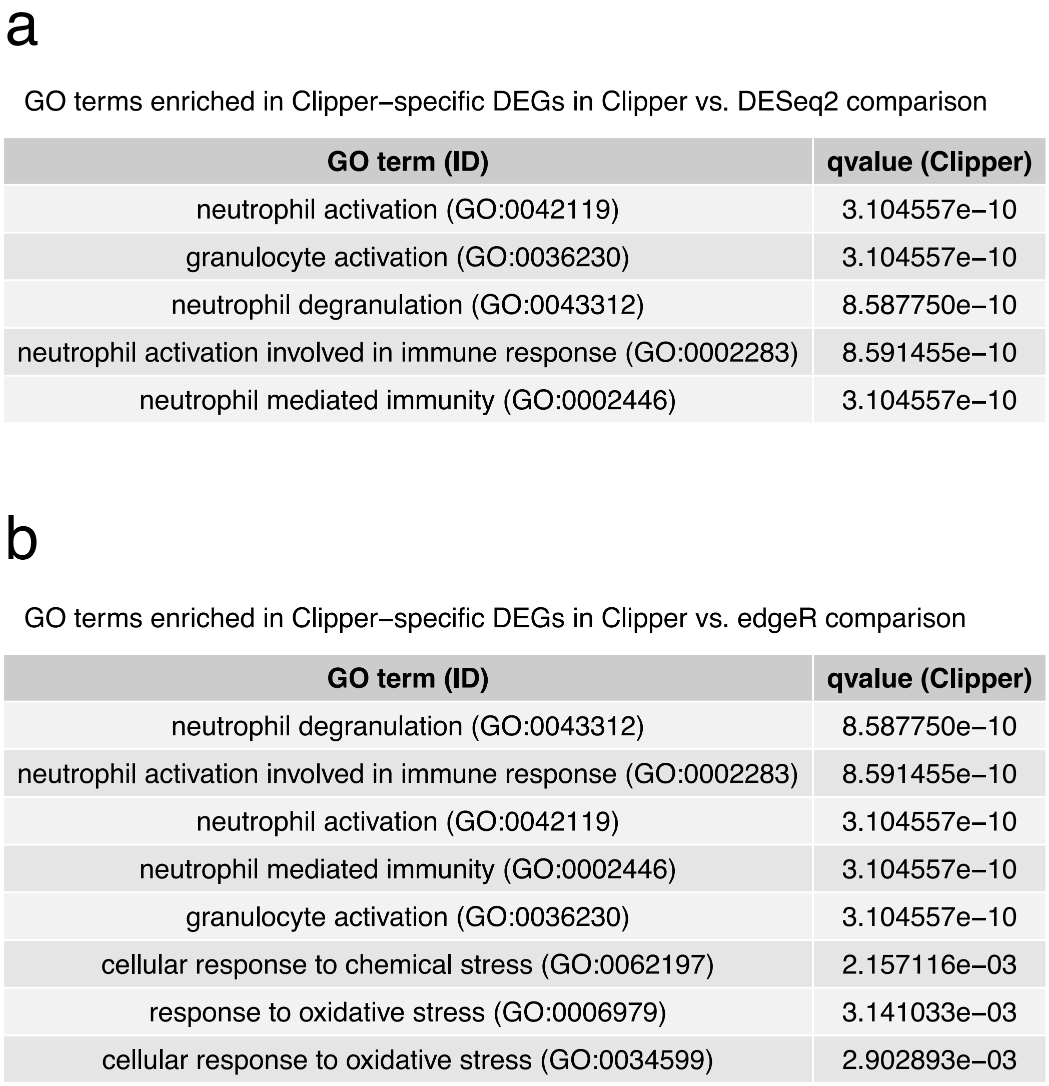


Figure 2.25: Comparison of Clipper and five scRNA-seq DEG identification methods on synthetic Drop-seq data generated by scDesign2 (based on a real Drop-seq dataset of PBMCs). The target FDR threhold q ranges from 1% to 10%.

In the "Actual FDR vs. Target FDR" plot (left), points above the dashed diagonal line indicate failed FDR control. Clipper controls the FDR while maintaining high power, demonstrating Clipper's good performance in single-cell DE analyses.



a

GO terms enriched in Clipper−specific DEGs in Clipper vs. DESeq2 comparison

GO term (ID)	qvalue (Clipper)
neutrophil activation (GO:0042119)	3.104557e−10
granulocyte activation (GO:0036230)	3.104557e−10
neutrophil degranulation (GO:0043312)	8.587750e−10
neutrophil activation involved in immune response (GO:0002283)	8.591455e−10
neutrophil mediated immunity (GO:0002446)	3.104557e−10

b

GO terms enriched in Clipper−specific DEGs in Clipper vs. edgeR comparison

GO term (ID)	qvalue (Clipper)
neutrophil degranulation (GO:0043312)	8.587750e−10
neutrophil activation involved in immune response (GO:0002283)	8.591455e−10
neutrophil activation (GO:0042119)	3.104557e−10
neutrophil mediated immunity (GO:0002446)	3.104557e−10
granulocyte activation (GO:0036230)	3.104557e−10
cellular response to chemical stress (GO:0062197)	2.157116e−03
response to oxidative stress (GO:0006979)	3.141033e−03
cellular response to oxidative stress (GO:0034599)	2.902893e−03

Figure 2.26: Enrichment q-values of GO terms that are found enriched in the DEGs that are uniquely identified by Clipper in pairwise comparison of **(a)** Clipper vs. edgeR and **(b)** Clipper vs. DESeq2. These GO terms are all related to immune response and thus biologically meaningful.

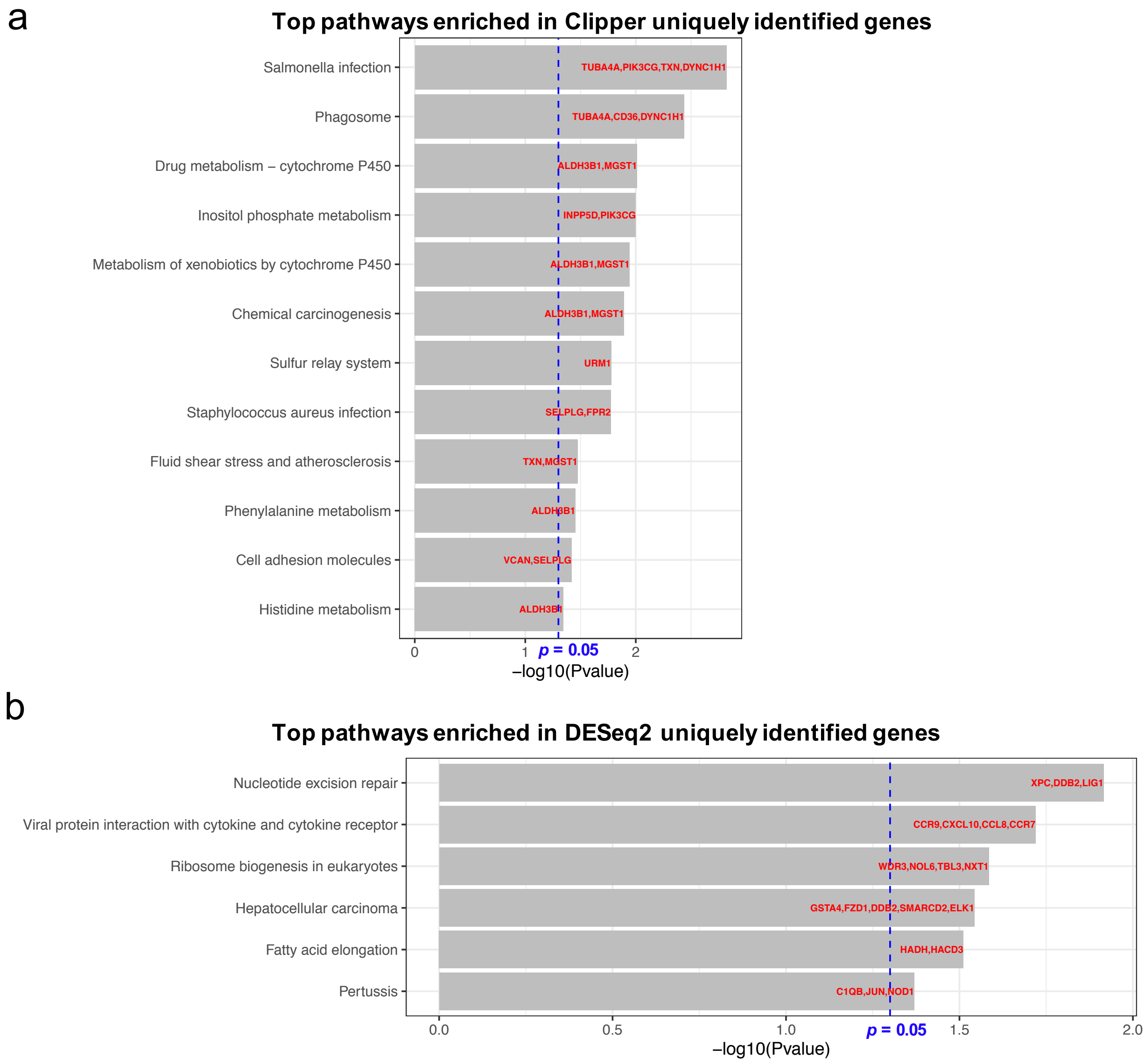


Figure 2.27: The p-values of the top enriched pathways in the DEGs that are uniquely identified by **(a)** Clipper and **(b)** DESeq2; i.e., the DEGs that are only identified by one method by missed by the other two methods. There are more immune-related pathways enriched in (a) than (b).

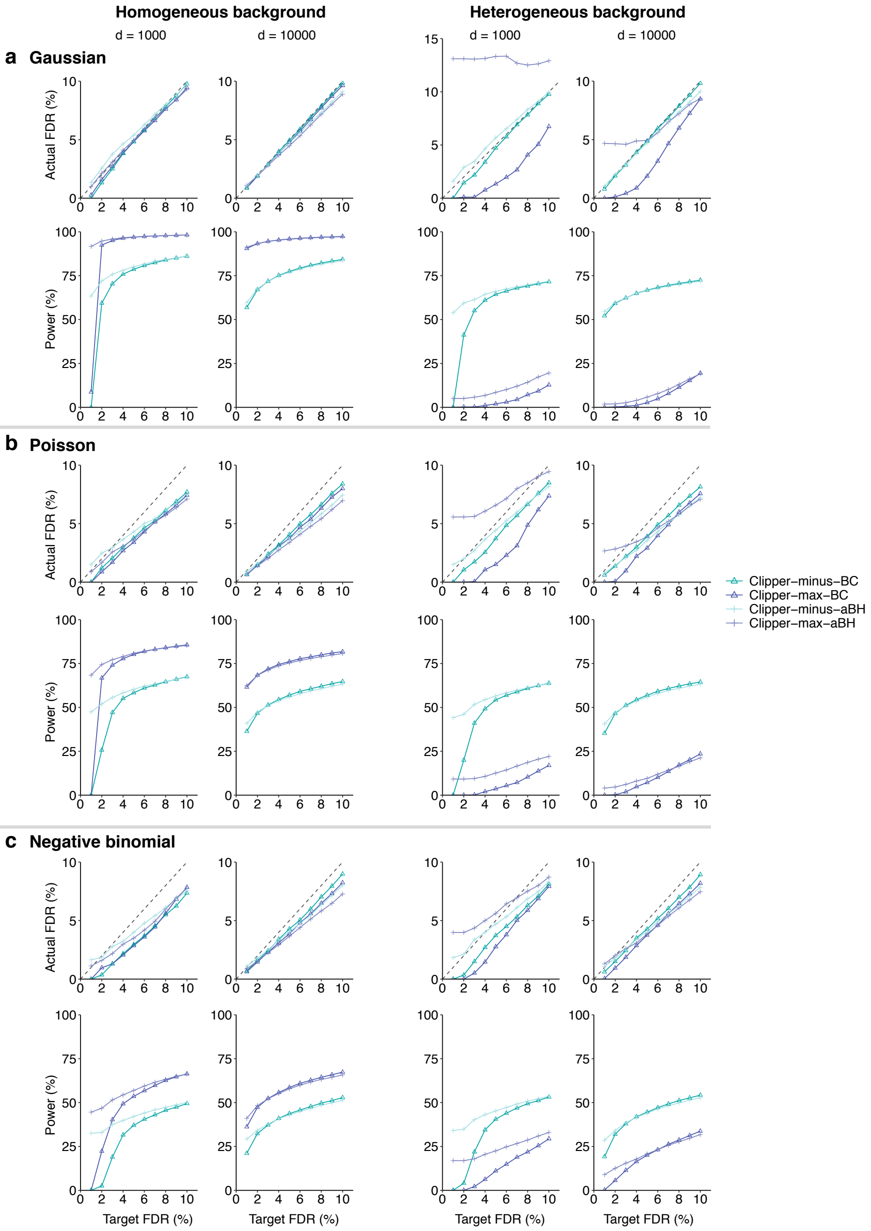


Figure 2.28: In 1vs1 enrichment analysis, comparison of four Clipper variant algorithms (Clipper-minus-BC, Clipper-minus-aBH, Clipper-max-BC, and Clipper-max-aBH) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 1000$ or 10,000 features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Clipper-minus-BC is chosen as the default implementation under this scenario.

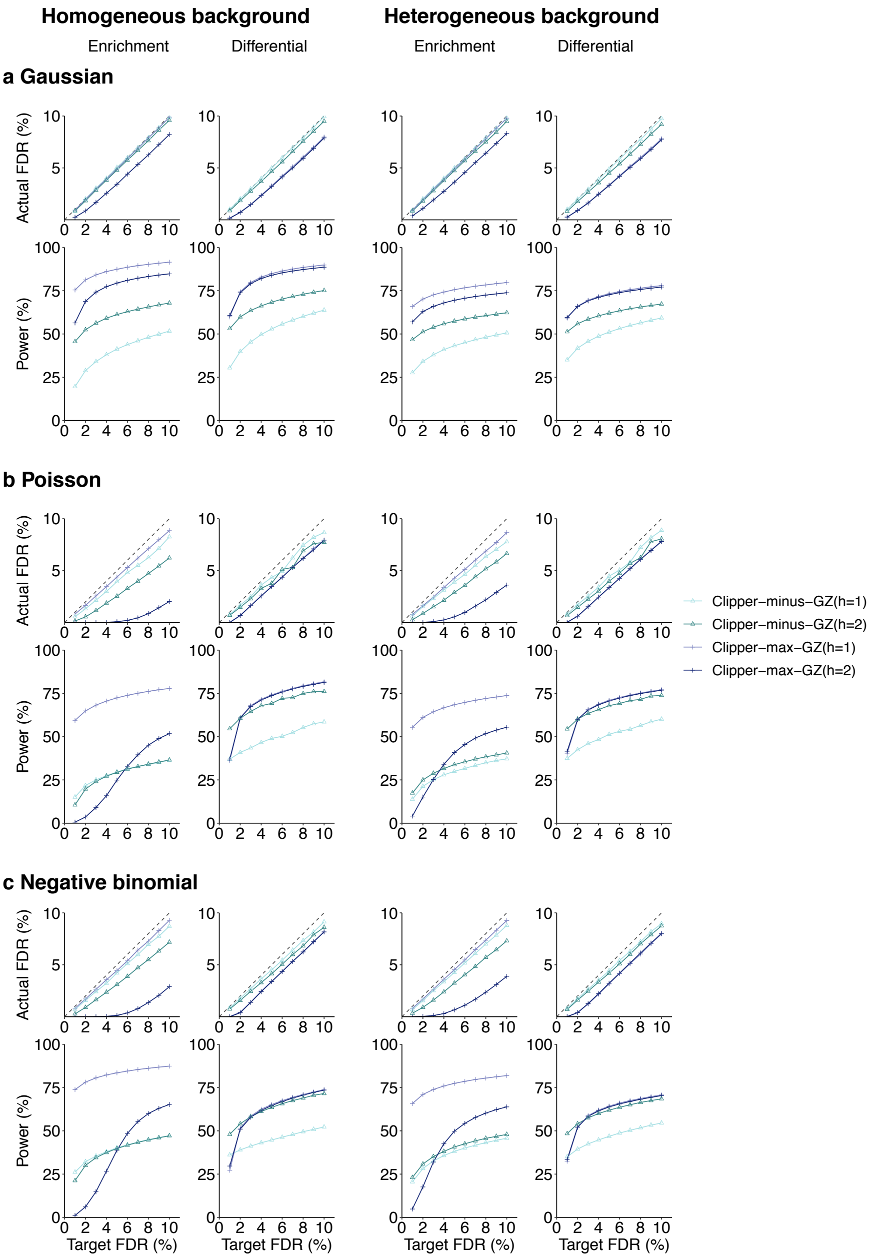


Figure 2.29: In the 2vs1 enrichment analysis (columns 1 and 3) and differential analysis (columns 2 and 4), comparison of four Clipper variant algorithms (Clipper-minus-GZ(h=1), Clipper-minus-GZ(h=2), Clipper-max-GZ(h=1), and Clipper-max-GZ(h=2)) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Clipper-max-GZ(h=1) is chosen as the default implementation under this scenario.

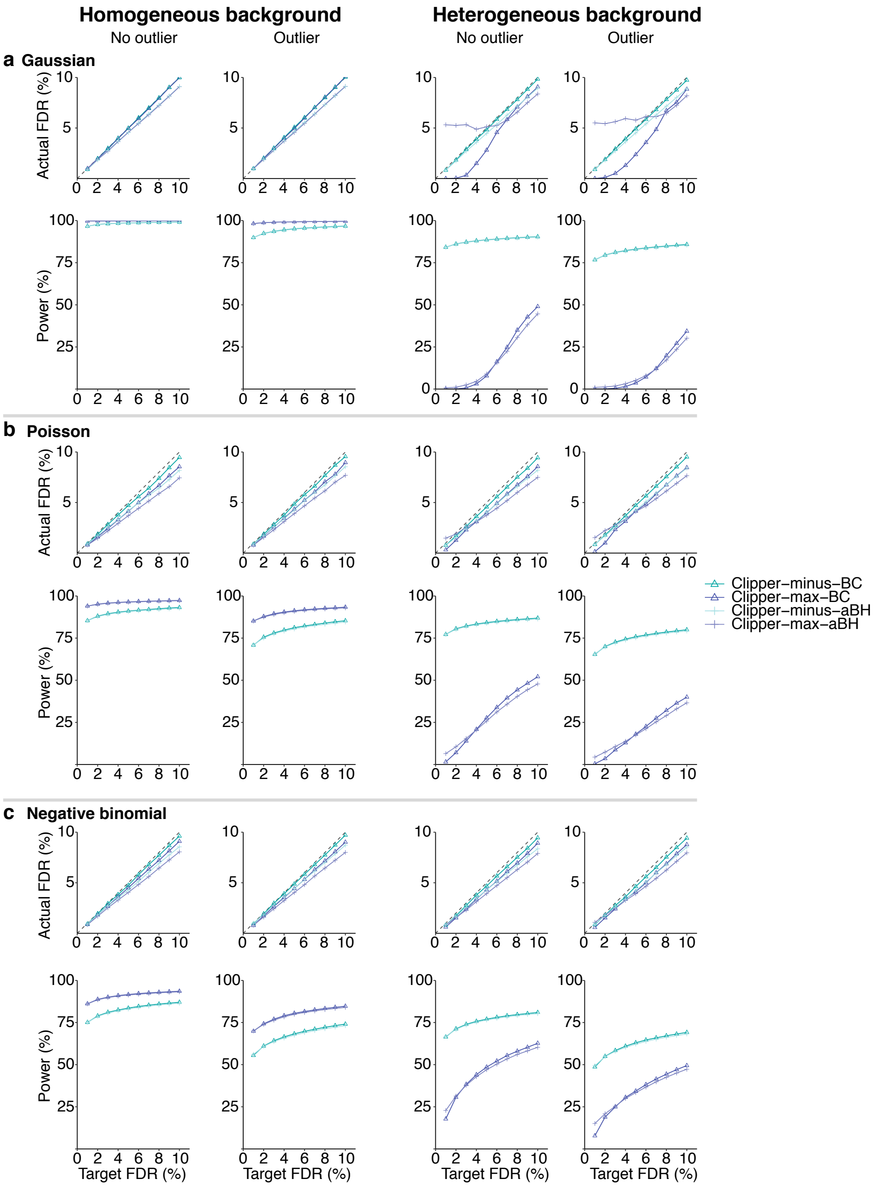


Figure 2.30: In 3vs3 enrichment analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of four Clipper variant algorithms (Clipper-minus-BC, Clipper-minus-aBH, Clipper-max-BC, and Clipper-max-aBH) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Clipper-minus-BC is chosen as the default implementation under this scenario.

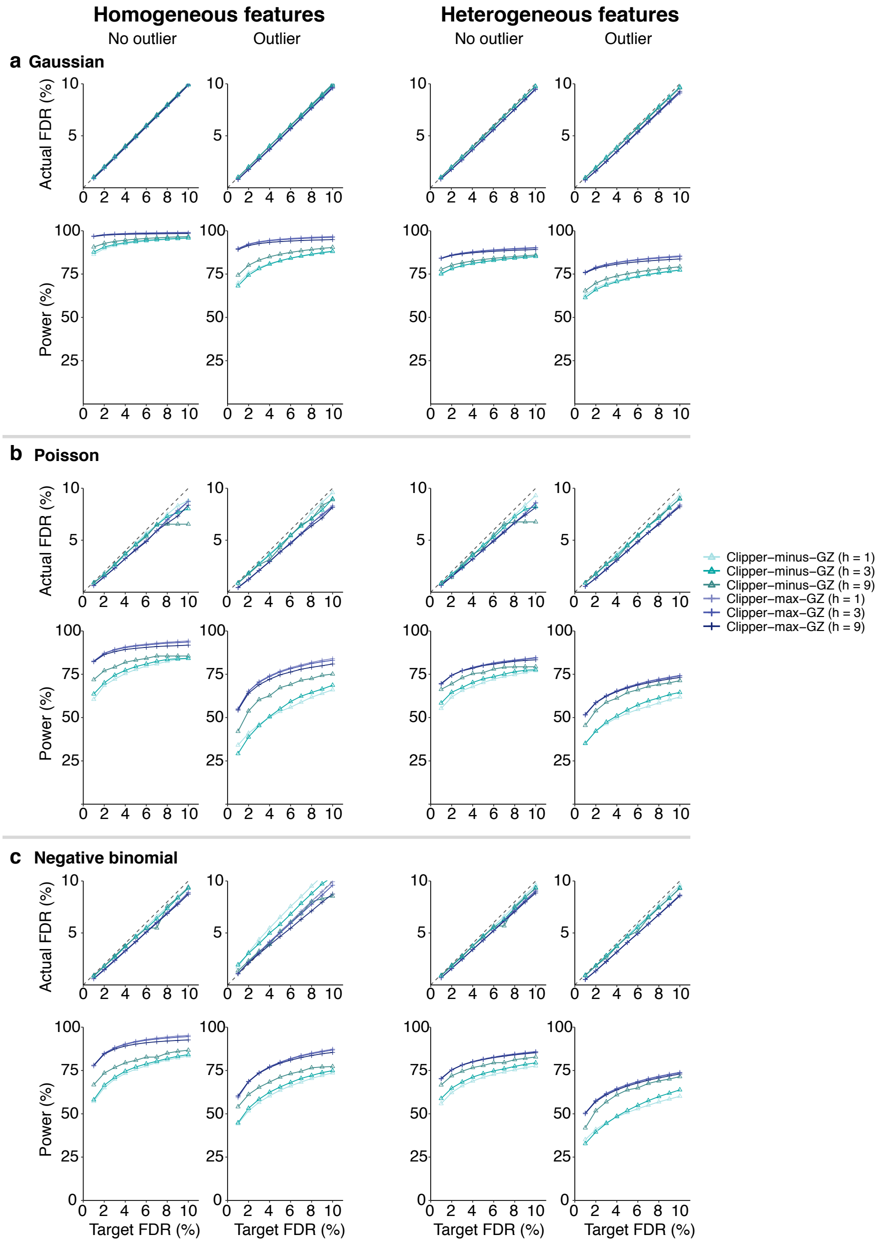


Figure 2.31: In 3vs3 differential analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of six Clipper variant algorithms (Clipper-minus-GZ(h=1), Clipper-minus-GZ(h=3), Clipper-minus-GZ(h=9), Clipper-max-GZ(h=1), Clipper-max-GZ(h=3), and Clipper-max-GZ(h=9)) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with d = 10,000 features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Clipper-max-GZ(h=1) is chosen as the default implementation under this scenario.

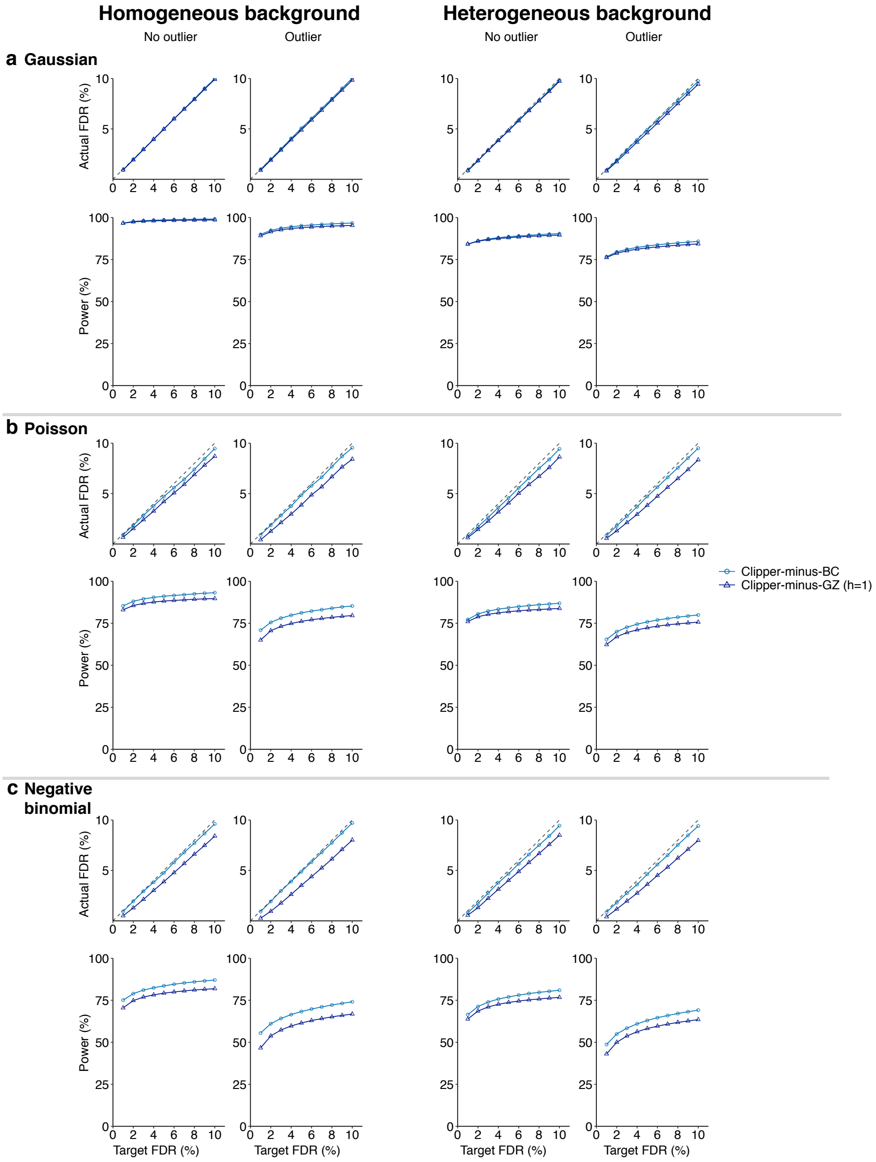


Figure 2.32: In the 3vs3 enrichment analysis without (columns 1 and 3) or with outliers (columns 2 and 4), comparison of two Clipper variant algorithms (Clipper-minus-BC, Clipper-max-GZ(h=1)) in terms of their FDR control and power.

At target FDR thresholds $q \in \{1\%, 2\%, \cdots, 10\%\}$, each method's actual FDRs and power are evaluated on 200 simulated datasets with $d = 10{,}000$ features generated from (**a**) the Gaussian distribution, (**b**) the Poisson distribution, or (**c**) the negative binomial distribution under homogeneous (two left columns) and heterogeneous (two right columns) background scenarios. Clipper-minus-BC is chosen as the default implementation under this scenario.

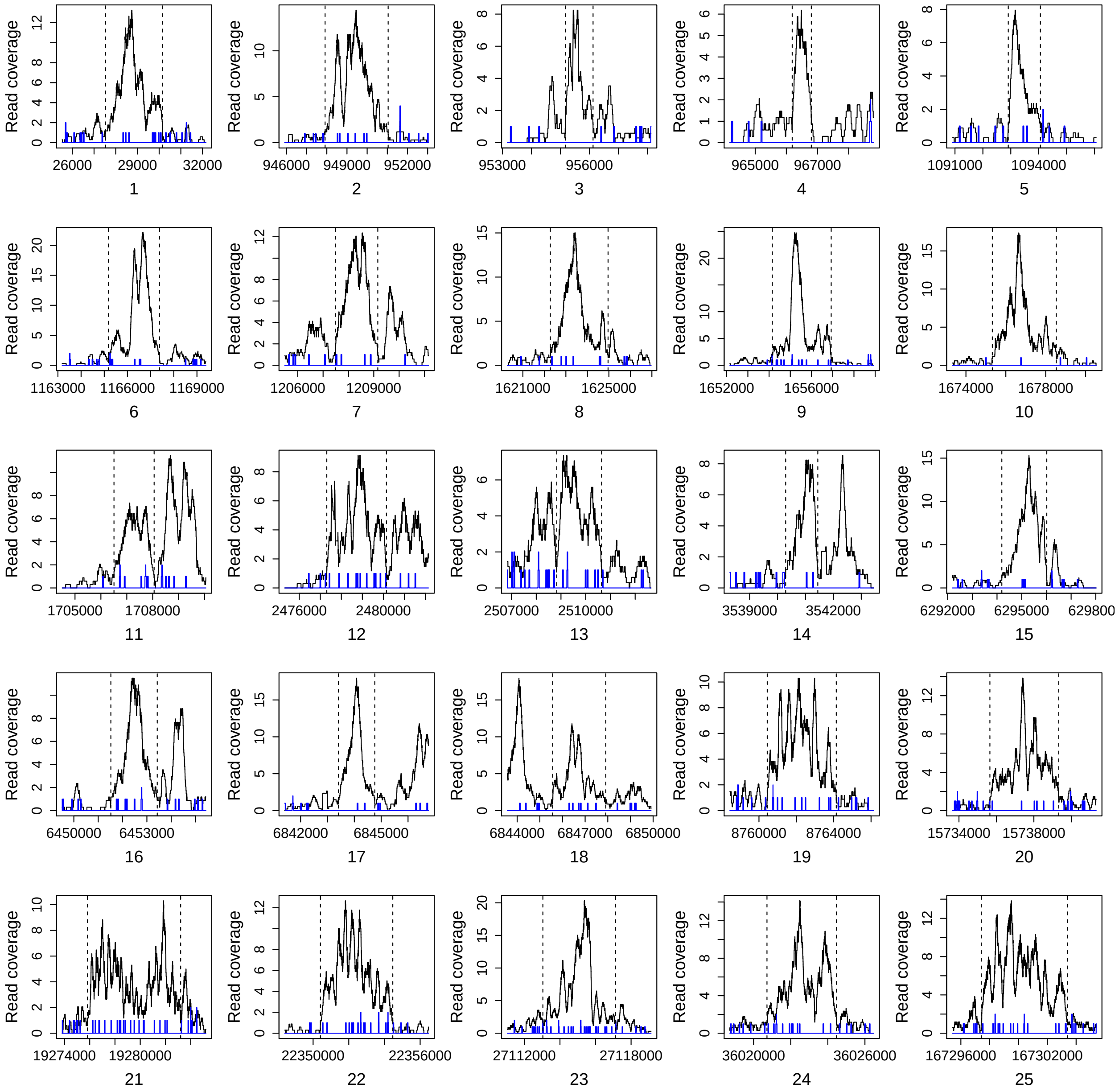


Figure 2.33: 25 "true peaks" from H3K4me3 ChIP-seq data of cell line GM12878. Black and blue curves indicate the read coverages in the experimental and control samples, respectively. Vertical dashed lines indicate the peak boundaries.

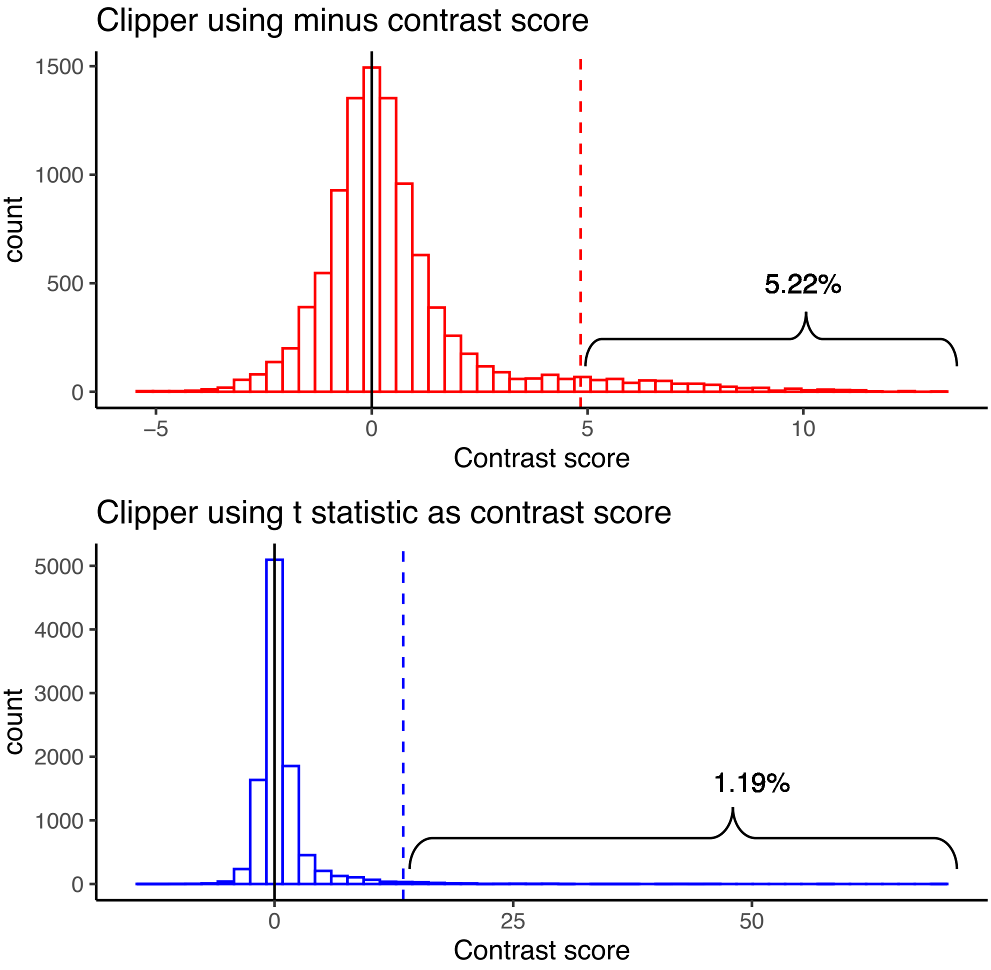


Figure 2.34: In the 3vs3 enrichment analysis, distributions of contrast scores used by two Clipper variants: the default Clipper using the minus contrast score (top) and the Clipper variant using the two-sample t statistic (bottom).

Features are generated from the Gaussian distribution under the heterogeneous background scenario (see Supp. Section S2.5.3). The vertical dashed lines indicate the contrast score cutoffs found by the BC procedure at the target FDR threshold $q = 1\%$. The distribution of the minus contrast scores has a heavier right tail (5.22%) than that of the distribution of the t statistic contrast scores (1.19%).

CHAPTER 3

An alignment-based bioinformatic tool for comparing chromatin state sequences from two conditions

3.1 Introduction

All tissue and cell types, such as embryonic stem cells (ESCs), terminally differentiated tissues, and cultured cell lines, are maintained and controlled by epigenomic regulation and gene expression programs [76–78]. An epigenome encodes information of chemical modifications to DNA and histone proteins of a genome, and such modifications may result in changes to chromatin structures and genome functions. Epigenomic information is represented by multi-track signals, including DNA methylation, covalent histone modifications, and DNA accessibility, all of which are measured genome-wide by high-throughput sequencing technologies such as Bisulfite-seq, ChIP-seq and DNase-seq [79]. In recent years, multiple international consortia, including the Encyclopedia of DNA elements (ENCODE) [57], the NIH Roadmap Epigenomics Mapping Consortium [20, 80], and the International Human Epigenome Consortium [81], have generated large-scale high-throughput epigenome sequencing datasets for a broad spectrum of tissue and cell types, offering an unprecedented opportunity for studying multiple levels of epigenetic regulation across diverse cell states. Specifically, the NIH Roadmap project has released public epigenomic data of 127 human tissue and cell types [20]. This database contains a total of 2,804 genome-wide epigenomic datasets, including 1,821 histone modification datasets, 360 DNase datasets, and 277 DNA methylation datasets.

A series of computational methods, including ChromHMM [14], Segway [15], GATE [82], TreeHMM [83], STAN [84], EpiCSeg [85], Spectacle [86], IDEAS [87], and GenoSTAN [88], have been developed to build a genome-wide chromatin state annotation, where distinct chromatin states have demonstrated diverse regulatory and transcriptional signals [16–18]. In these methods, each epigenome is segmented into non-overlapping regions, and a single-track chromatin state sequence is constructed by compressing the multi-track epigenetic activities (e.g., DNA methylation and histone modifications) in various ways. For example, ChromHMM assigns discrete chromatin state labels to genomic regions based on signals of multiple epigenetic markers using a hidden Markov model [14]. The predicted chromatin states have shown strong biological relevance and wide applicability in genomic research, e.g., the identification of enhancers and promoters [18]. Given a chromatin state annotation constructed by any of these methods, genomic regions of the same chromatin state are expected to have both consistent epigenomic patterns and similar regulatory functions.

Based on existing chromatin state annotations, previous work has studied similarities and differences of human tissue and cell types in terms of epigenomic signals in specific functional genomic elements (e.g., promoters and enhancers), as well as the tissue and cell specificity of these elements, using the Pearson correlation coefficients [19, 20] or a newly developed epigenome overlap measure (EPOM) [21]. The aforementioned methods have shed significant insights into our understanding of gene regulation on a global scale, i.e., how promoters and enhancers regulate target genes in diverse tissue and cell types. However, former epigenome comparative studies failed to effectively incorporate the sequential information of chromatin states, which, however, we believe are highly likely to contain critical information on gene regulatory mechanisms.

The comparison of DNA/RNA or protein sequences is based on the sequential information of nucleotides or amino acids. Many sequence alignment methods have been developed over the past decades to measure the similarity between sequences. Earlier work such as the Needleman-Wunsch algorithm [22] and the Smith-Waterman algorithm [23] use dynamic programming to search for the best global or local matches between two sequences. With the development of these algorithms, sequence alignment tools have become indispensable in almost all modern biological research. They are powerful not only in studies that focus on comparing sequences, such as evolutionary studies, but also in query-database retrieval studies, which aim to find regions from a large database that are similar to the query sequence of interest. However, there is no alignment algorithm designed to assess the epigenetic similarity of long genomic regions, such as gene regions and long non-coding regulatory regions. A main challenge lies in the multi-track nature of epigenomic signals. On the one hand, substantial information would be lost if we calculate a scalar value (e.g., the mean signal averaged over multiple 25 bp windows) to represent the signal of a long genomic region per track per tissue/cell. On the other hand, if we directly analyze the original data (a signal value per 25 bp window per track per tissue/cell), we would need to evaluate the similarity of large matrices to compare genomic regions. Specifically, the matrix of a region has the dimensions as the number of 25 bp windows in the region × the number of tracks. Given that different regions almost certainly have different region lengths thus they have matrices of different dimensions, how to evaluate their similarity is a non-trivial task. In addition, we also need to consider the fact that a long region often contains multiple functional genomic elements with varying lengths. Hence, a reasonable approach is to compare two long regions based on their chromatin state patterns learned from multiple-track epigenomic signals. Motivated by the fact that chromatin state sequences provide a biologically meaningful one-track interpretation of multi-track epigenomic signals [14], we reduce the challenging question of comparing long multi-track epigenomic signals to a simpler task of comparing two chromatin state sequences.

Given the fast accumulation of large-scale epigenomic datasets generated in recent years, biological researchers are in great need of a new bioinformatic tool to efficiently retrieve genomic regions similar to an interested query region in terms of epigenomic signals. Motivated by the enormous successes of sequence alignment algorithms in comparing nucleotide and protein sequences [24], here we propose a novel computational method, Epigenome Alignment (EpiAlign), to compare two genomic regions by aligning their chromatin state sequences. To the best of our knowledge, EpiAlign is the first pairwise alignment-based method that investigates the sequential patterns of chromatin states and studies the epigenome similarity based on the patterns. EpiAlign compares two chromatin state sequences by calculating a local alignment score. It also allows the search of genomic regions (i.e., "hits") whose chromatin state sequences are similar to those of a query region. Aligned chromatin state sequences are expected to have similar biological functions. EpiAlign is flexible in performing the chromatin state sequence alignment either within an epigenome, i.e., a tissue or cell, or between two epigenomes. From the alignment results of EpiAlign, users can identify common chromatin state patterns to investigate the functional relationship of interested genomic regions.

3.2 EpiAlign methodology

The EpiAlign algorithm aims to find an optimal local alignment between two chromatin state sequences. Our algorithm development is motivated by the classic Smith-Waterman Algorithm [23]. We design the mismatch and deletion score functions based on the weight of each chromatin state in each sequence. We first apply a chromatin state annotation method (e.g. ChromHMM [14]) to encode multi-track epigenomic signals into single-track chromatin state sequences, whose different states are represented by different labels. Second, we compress consecutive occurrences of the same state into a state label. For example, a chromatin state sequence `abbcc` is represented by a compressed state sequence S = `abc`. EpiAlign then performs a local alignment between two genomic regions based on their compressed state sequences. The motivation of adding a compression step lies in the fact that most uncompressed (chromatin state) sequences contain long stretches of a single chromatin state, mostly the quiescent/low state (see Supplementary section 2), and including such length information would dominate the alignment result, a scenario that is often undesirable, because the purpose of alignment is to find similar chromatin state patterns composed of more than one state. The compression step allows EpiAlign to focus more on chromatin state patterns instead of a single chromatin state that spans a long genomic region. We use an example to demonstrate the effectiveness of adding the compression step to address this issue: in the brain sample E071, when we applied EpiAlign with the compression step, the brain-specific gene *NRG3* has the best alignment with another brain-specific gene *GRIA1*, among all the protein-coding genes. This result is reasonable as both genes are brain-specific and highly expressed in brain samples. However, as these two genes have vastly different lengths (*NRG3* is three times longer than *GRIA1*) and their chromatin state sequences have long stretches of the quiescent/low state, they are poorly aligned when we applied EpiAlign without the compression step. This result indicates that the compression step, which condenses the epigenetic information encoded in chromatin state sequences, is necessary and effective for finding similar and biologically meaningful chromatin state patterns. Additionally, aligning uncompressed sequences is much more time-consuming (20 times more computation time on average) than aligning their compressed counterparts. Therefore, adding the compression step also increases the computational efficiency of EpiAlign. In the following text, unless specified, all the chromatin state sequences refer to the compressed state sequences.

3.2.1 Modified Smith-Waterman Algorithm for Chromatin State Sequence Alignment

Given two chromatin state sequences S_1 and S_2, we characterize a possible alignment between S_1 and S_2 through a set of triplets $\{(f_i, u_{1i}, u_{2i})\}_{i=1}^{N}$, where N denotes the total number of aligned basepairs (including matches, mismatches, and gaps), f_i gives the alignment status between two chromatin states whose positions are u_{1i} and u_{2i} in S_1 and S_2, respectively. We may equivalently write this set of triplets as three equal-length sequences: $F = f_1 f_2 \cdots f_N$, $U_1 = u_{11}u_{12} \cdots u_{1N}$, and $U_2 = u_{21}u_{22} \cdots u_{2N}$. Specifically, $f_i \in \{\mathtt{m}, \mathtt{n}, \mathtt{d}_1, \mathtt{d}_2\}$ denotes one of the four possible alignment status between two chromatin states: $\mathtt{m}$ for match, $\mathtt{n}$ for mismatch, $\mathtt{d}_1$ for deletion in S_1, and $\mathtt{d}_2$ for deletion in S_2. If $f_i = \mathtt{m}$, there is a match between the u_{1i}-th state of S_1 and the u_{2i}-th state of S_2; if $f_i = \mathtt{n}$, there is a mismatch between the u_{1i}-th state of S_1 and the u_{2i}-th state of S_2; if $f_i = \mathtt{d}_1$, the u_{1i}-th state of S_1 is aligned to nothing in S_2 (u_{2i} is set to 0); if $f_i = \mathtt{d}_2$, the u_{2i}-th state of S_2 is aligned to nothing in S_1 (u_{1i} is set to 0). In an example with $S_1 = \mathtt{abca}$ and $S_2 = \mathtt{aba}$, if we consider an alignment $\begin{array}{cccc} \mathtt{a} & \mathtt{b} & \mathtt{c} & \mathtt{a} \\ | & | & | & | \\ \mathtt{a} & \mathtt{b} & - & \mathtt{a} \end{array}$, then $F = \mathtt{mmd_1m}$, $U_1 = \mathtt{1234}$, and $U_2 = \mathtt{1203}$. Please note that the two chromatin state sequences S_1 and S_2 may have different lengths. Also given S_1 and S_2, it is possible to have more than one alignment results, i.e., sets of $\{(f_i, u_{1i}, u_{2i})\}_{i=1}^{N}$.

Now we define the alignment score function $H(\cdot)$ as:

$$H(F, U_1, U_2, S_1, S_2) = \sum_{i=1}^{N} h(f_i, u_{1i}, u_{2i}, S_1, S_2)\,, \tag{3.1}$$

where $h(f_i, u_{1i}, u_{2i}, S_1, S_2)$ denotes the score of the alignment status f_i between the u_{1i}-th state in S_1 and the u_{2i}-th state in S_2. Specifically,

- $h(\mathtt{m}, u_{1i}, u_{2i}, S_1, S_2) = \mathrm{MF}(u_{1i}, u_{2i}, S_1, S_2)$;
- $h(\mathtt{n}, u_{1i}, u_{2i}, S_1, S_2) = \mathrm{NF}(u_{1i}, u_{2i}, S_1, S_2)$;
- $h(\mathtt{d}_1, u_{1i}, u_{2i}, S_1, S_2) = \mathrm{DF}(u_{1i}, S_1)$;
- $h(\mathtt{d}_2, u_{1i}, u_{2i}, S_1, S_2) = \mathrm{DF}(u_{2i}, S_2)$.

We will formally define the matching function $\mathrm{MF}(\cdot)$, the mismatching function $\mathrm{NF}(\cdot)$, and the deletion function $\mathrm{DF}(\cdot)$ later in this section. To summarize, the function $h(\cdot)$ takes a form that depends on the value of its first argument f_i.

Then we consider the alignment problem as an optimization problem where the goal is to find the optimal alignment $\{F^*, U_1^*, U_2^*\}$ that maximizes the alignment score H:

$$\{F^*, U_1^*, U_2^*\} = \operatorname*{arg\,max}_{\{F, U_1, U_2\}} H(F, U_1, U_2, S_1, S_2)\,. \tag{3.2}$$

This optimization problem can be approached by dynamic programming, an algorithm that iteratively maintains and updates a matrix M that stores dynamic alignment results. The matrix element $M_{k,l}$ is the maximal alignment score of the two subsequences $S_1^{[1,k]}$ and $S_2^{[1,l]}$, where $S_1^{[1,k]}$ denotes the first k states of S_1 and $S_2^{[1,l]}$ denotes the first l states of S_2. Let n_1 and n_2 be the length of S_1 and S_2, respectively. We update the matrix M using the following rule.

$$\begin{aligned}
M_{k,0} &= 0\,, \;\; \text{for } 0 \leqslant k \leqslant n_1\,; \\
M_{0,l} &= 0\,, \;\; \text{for } 0 \leqslant l \leqslant n_2\,; \\
M_{k,l} &= \max \begin{cases} M_{k-1,l-1} + \mathrm{MF}\,(k,l,S_1,S_2) & \text{Match} \\ M_{k-1,l-1} + \mathrm{NF}\,(k,l,S_1,S_2) & \text{Mismatch} \\ M_{k-1,l} + \mathrm{DF}(k,S_1) & \text{Deletion in } S_1 \\ M_{k,l-1} + \mathrm{DF}(l,S_2) & \text{Deletion in } S_2 \end{cases}, \\
&\quad \text{for } 1 \leqslant k \leqslant n_1,\; 1 \leqslant l \leqslant n_2.
\end{aligned} \tag{3.3}$$

The algorithm described in Equation (3.3) achieves the global alignment, but we instead consider the local alignment approach in practice since the local alignment would prefer long continuous alignments with small proportion of mismatches, which are more likely to contain the common patterns of interest. In contrast, global alignment would prefer patterns containing overly scattered short alignments separated by gaps. To achieve the goal of local alignment, we propose the following approach to modify the dynamic programming algorithm.

$$\begin{aligned}
M_{k,0} &= 0\,, \;\; \text{for } 0 \leqslant k \leqslant n_1\,; \\
M_{0,l} &= 0\,, \;\; \text{for } 0 \leqslant l \leqslant n_2\,; \\
M_{k,l} &= \max \begin{cases} 0 & \\ M_{k-1,l-1} + \mathrm{MF}\,(k,l,S_1,S_2) & \text{Match} \\ M_{k-1,l-1} + \mathrm{NF}\,(k,l,S_1,S_2) & \text{Mismatch} \\ M_{k-1,l} + \mathrm{DF}(k,S_1) & \text{Deletion in } S_1 \\ M_{k,l-1} + \mathrm{DF}(l,S_2) & \text{Deletion in } S_2 \end{cases}, \\
&\quad \text{for } 1 \leqslant k \leqslant n_1,\; 1 \leqslant l \leqslant n_2.
\end{aligned} \tag{3.4}$$

The alignment score of EpiAlign is $M^{\text{EpiAlign}} = M_{n_1,n_2}$.

3.2.2 Chromatin State Weights

To define the specific forms of the matching function $\mathrm{MF}(\cdot)$, the mismatching function $\mathrm{NF}(\cdot)$ and the deletion function $\mathrm{DF}(\cdot)$, we first introduce a weight function $W(k, S)$, which describes the weight of the k-th state in sequence S. The weights can be used to distinguish chromatin states of different importance if we have prior knowledge that some states have more significant biological functions than others at certain positions. We design two sets of weights: (1) **equal weights** mean that all states are treated equally with the same weight 1 in sequence S, i.e., $W(k, S) = 1\,,\ k = 1, \ldots, |S|$; (2) **frequency-based weights** assign larger weights to less common chromatin states (see Supplementary section 1 for details), motivated by the fact that some uncommon states are likely strong indicators of biological functions.

With the weights defined above, we specify the matching function, the mismatching function, and the deletion function as:

$$\mathrm{MF}(k, l, S_1, S_2) = W(k, S_1) + W(l, S_2)\,, \tag{3.5}$$

$$\mathrm{NF}(k, l, S_1, S_2) = -\epsilon_N \cdot (W(k, S_1) + W(l, S_2))\ , \tag{3.6}$$

$$\mathrm{DF}(k, S) = -\epsilon_D \cdot W(k, S)\,, \tag{3.7}$$

where ϵ_N and ϵ_D are the penalty parameters for a mismatch and a deletion in the alignment, respectively. In EpiAlign, ϵ_N and ϵ_D can be tuned by users, and the default values are 1.5 and 1, respectively. The choice of ϵ_N and ϵ_D values depends on how "local" users would like the result to be, i.e., if we set a larger ϵ_N or ϵ_D value, it means that we penalize more on a mismatch or a gap in the alignment, and the final best alignment result will be shorter or more local. Figure 3.1 shows the workflow of EpiAlign

3.3 Results

We demonstrate in three aspects that EpiAlign is a useful tool for investigating sequential patterns of chromatin states. First, we demonstrate that EpiAlign can identify common

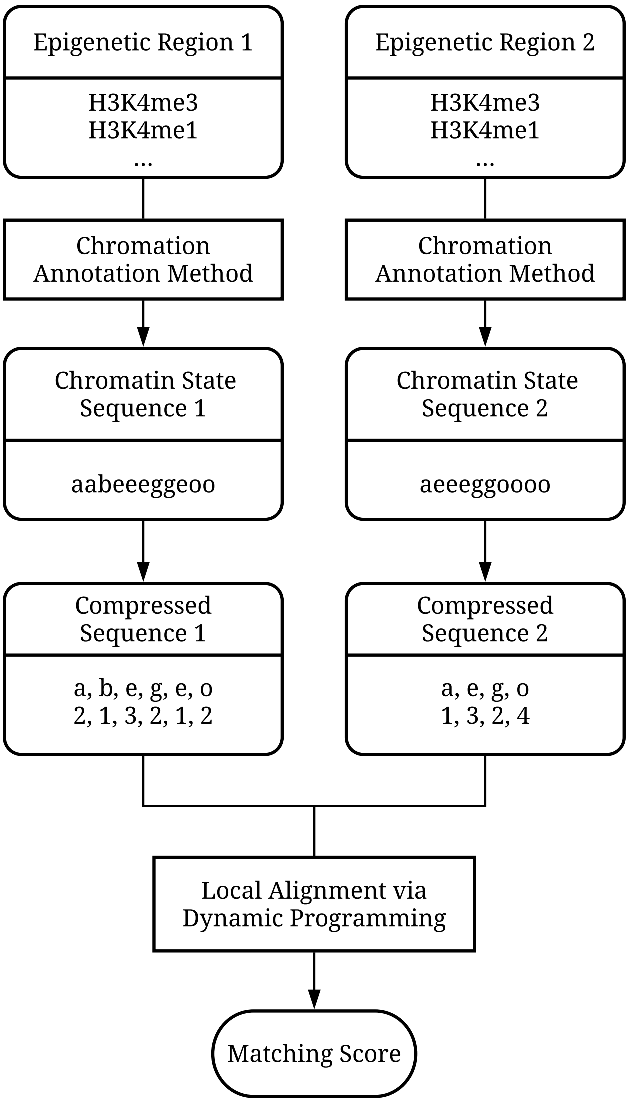


Figure 3.1: Workflow of the EpiAlign algorithm.

chromatin state patterns within the same epigenome or across different epigenomes. Second, we investigate biological interpretation of the common chromatin state patterns found by EpiAlign. Third, as a technical verification, we show that EpiAlign is able to distinguish real epigenomes from randomized epigenomes. We also demonstrate the superiority of EpiAlign over a naïve method that compares two chromatin sequences only based on chromatin state frequencies. We conduct the above analysis using simulation and real case studies based on the Roadmap epigenomic database [20]. In this paper, we use the chromatin state sequences annotated by ChromHMM, which has been well recognized to provide an informative compression of multi-track epigenomic signals into a chromatin state sequence [14, 20, 21]. It is worth noting that our method is generally applicable to chromatin state sequences annotated by other methods.

In this paper, for most analysis, we selected ESC, heart and brain samples from the Roadmap dataset as representative examples. The reason is that among all the Roadmap tissue types, these three types are relatively better understood and have well-annotated tissue-specific genes[89].

3.3.1 Vertical alignment: Comparison of Chromatin State Sequences of Protein-coding Genes across Epigenomes

EpiAlign is a powerful local alignment algorithm to quantify the similarity of two chromatin state sequences in terms of their aligned subsequences. Here we apply EpiAlign to compare chromatin state sequences of the same genomic region in different epigenomes, a strategy we define as the **vertical alignment**. The diversity of the same region's chromatin state sequences represents epigenetic characteristics of various tissues and cell types. As epigenetic characteristics are known to have a strong association with gene expression characteristics [90], we expect that a cell-type specific gene, i.e., a gene specifically highly expressed in a cell type [89], should have similar chromatin state sequences in epigenomes of that cell type. In contrast, lower similarity is expected between two chromatin state sequences, one of that cell type and the other of another cell type (Supplementary Figures 3 and 4).

In the first study, we divide the Roadmap epigenomes into two categories: 51 male samples and 38 female samples. In the second study, we compare the Roadmap epigenomes of two cell types: 10 brain samples and 5 heart samples. In both studies, we compare the chromatin state sequences for each of the $19,935$ protein-coding genes between every pair of samples. (Note that we use all protein-coding genes in GENCODE v10 [91] that are compatible with the Roadmap database, with the exception of genes on chromosome Y.)

We obtain three sets of alignment scores: pairwise scores within male samples, pairwise scores between male and female samples, and pairwise scores within female samples. Since most genes on the X chromosome are associated with sex-linked traits, we expect to observe higher alignment scores between samples of the same sex than those between samples of different sexes. To quantify the difference between alignment scores, we perform the two-sample one-sided Wilcoxon test between male-vs-male scores and male-vs-female scores for each protein-coding gene. Studying the resulting p-values, we find that out of the top 200 genes that have the smallest p-values, 188 are X chromosome genes. (Figure 3.2(a)). This result suggests that the majority of the genes that exhibit greater within-sex similarity are sex linked, a reasonable finding that matches our expectation. The comparison between female-vs-female and male-vs-female alignment scores leads to a similar result (Figure 3.2(b)). These results together confirm that EpiAlign successfully distinguishes same-sex chromatin state sequences from different-sex ones, suggesting that EpiAlign outputs a reasonable similarity measure of chromatin state sequences.

We also investigate the 12 genes that are not on X chromosome among the top 200 genes with the smallest p-values (Supplementary Table 1). These genes are potentially sex linked. For example, *MFF* that controls mitochondrial fission has been reported to have to have sex-specific regulation [92]. This result suggests that EpiAlign can serve as a useful tool for discovering genomic regions with certain epigenetic regulation of interest.

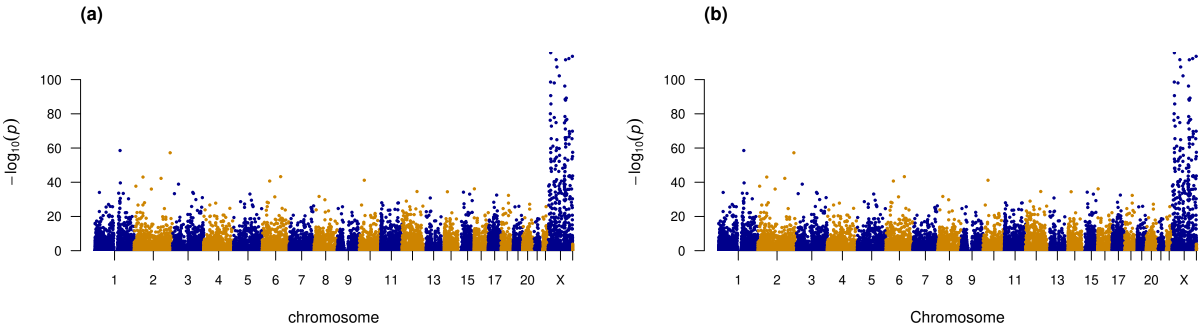


Figure 3.2: Alignment scores of chromatin state sequences of protein-coding genes within a sex vs. between sexes.

We perform the two-sample one-sided Wilcoxon test between within-sex alignment scores and between-sex scores to quantify their differences: (a) Manhattan plot of p-values of the test between male-vs-male and male-vs-female alignment scores for all the protein-coding genes. (b) Manhattan plot of p-values of the test between female-vs-female and male-vs-female alignment scores for all the protein-coding genes. In the two comparisons, within-sex and between-sex alignment scores differ most significantly for genes on the X chromosome.

In the second study, we investigate if EpiAlign can help identify cell-type specific genes, which were previously discovered from gene expression profiles [89], using only chromatin state sequences. We perform the two-sample one-sided Wilcoxon test between brain-vs-brain alignment scores and brain-vs-heart alignment scores for all the $19,935$ protein-coding genes. We next perform the Gene Ontology (GO) enrichment analysis [93] on the top 200 genes that receive the smallest p-values in the Wilcoxon test (Supplementary Table 2). Here we choose the top 200 genes instead of setting a threshold on multiple-testing-adjusted p-values, because we found that the most commonly used threshold 0.05 led to a large number of significant genes. For our purpose of verifying that the top differentially aligned genes are biologically meaningful, choosing a smaller number of top ranked genes is a more reasonable approach. The top enriched GO terms (p-value < 0.0001) are highly relevant to heart/cardiac processes and brain processes (Table 1). Previously discovered 150 heart-specific genes and 166 brain-specific genes [89] are enriched in the top differential genes found by the Wilcoxon test, which have significantly higher within-tissue alignment scores than between-tissue scores. For example, 9 brain-specific genes and 4 heart-specific genes are in the top 100 differential genes (p-values $< 10^{-30}$ in a hyper-geometric test). Figure 3.3 shows that top differential genes contain a higher proportion of tissue-specific genes. The above results indicate that EpiAlign is able to distinguish cell-type specific genes by assigning them higher alignment scores when comparing the epigenomes of their associated cell types. This again suggests that EpiAlign effectively captures chromatin state patterns in epigenomes.

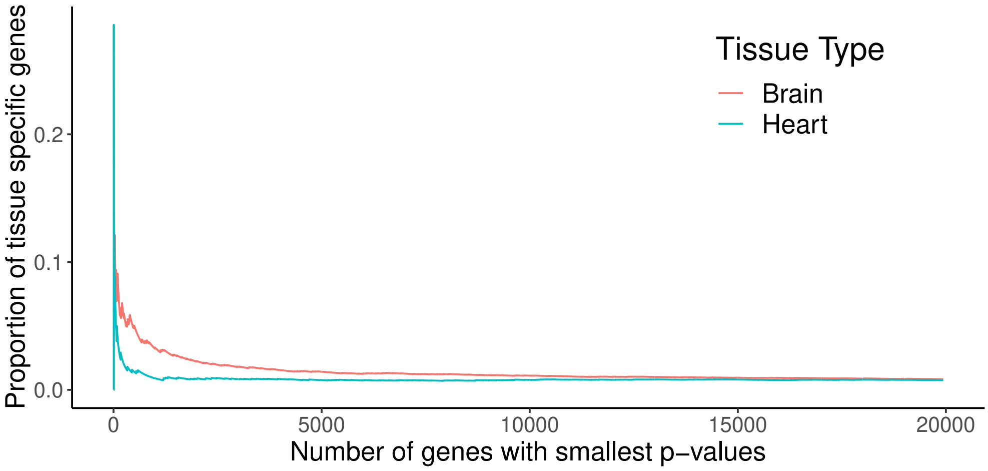


Figure 3.3: Brain and heart specific genes are enriched in the top differential genes that have significantly higher within-tissue alignment scores than between-tissue scores. The horizontal axis shows the number of top differential genes, and the vertical axis shows the proportion of tissue specific genes among the top differential genes.

GO term	Description	P-value
GO:0051891	*positive regulation of cardioblast differentiation	9.34E-8
GO:0051890	*regulation of cardioblast differentiation	6.42E-7
GO:0007416	**synapse assembly	5.82E-6
GO:0003207	*cardiac chamber formation	5.83E-6
GO:0060413	*atrial septum morphogenesis	1.72E-5
GO:0006928	movement of cell or subcellular component	2.15E-5
GO:0007409	**axonogenesis	2.98E-5
GO:0071625	vocalization behavior	3.07E-5
GO:0032990	cell part morphogenesis	4.63E-5
GO:2000738	positive regulation of stem cell differentiation	6.36E-5
GO:0060043	*regulation of cardiac muscle cell proliferation	6.99E-5
GO:0097104	**postsynaptic membrane assembly	8.69E-5
GO:0048812	**neuron projection morphogenesis	8.79E-5
GO:0051705	multi-organism behavior	9.73E-5

Table 3.1: Alignment scores of chromatin state sequences of protein-coding genes within a tissue (heart or brain) vs. between heart and brain. Displayed are the enriched GO terms in the top 200 significant genes identified by the Wilcoxon test between brain-vs-brain alignment scores and brain-vs-heart alignment scores. The top enriched GO terms are highly relevant to heart processes or brain processes (*: terms related with heart; **: terms related with brain).

To better illustrate how EpiAlign helps identify common chromatin state patterns, we study a brain-specific gene *STMN4*, which has the lowest p-value from our two-sample one-sided Wilcoxon test described above (brain-brain alignment scores vs. brain-heart alignment scores). Using it an example, we investigate the chromatin state sequences of *STMN4* in all brain and heart samples. From Figure 4, we observe that the brain samples share similar chromatin sequences; yet the common pattern in these sequences drastically differs from the chromatin state sequences in the heart samples. The fact that EpiAlign captured *STMN4* as the top differentially aligned gene shows that EpiAlign can successfully identify regions where chromatin state patterns diverge or conserve between cell types.

We also analyze the expression profiles of protein-coding genes. We use DESeq2 [29] and EdgeR [28] to do differential expression (DE) analysis between heart samples and brain samples on all the $17,784$ protein-coding genes included in the Roadmap RNA-seq datasets. The results show a high consistency between the resulting differentially expressed genes and the differential chromatin state sequences found by EpiAlign (Table 2). This results further validate that the tissue-specific regions found by EpiAlign are biologically meaningful and reflect gene expression dynamics, and that EpiAlign will be a useful tool for identifying

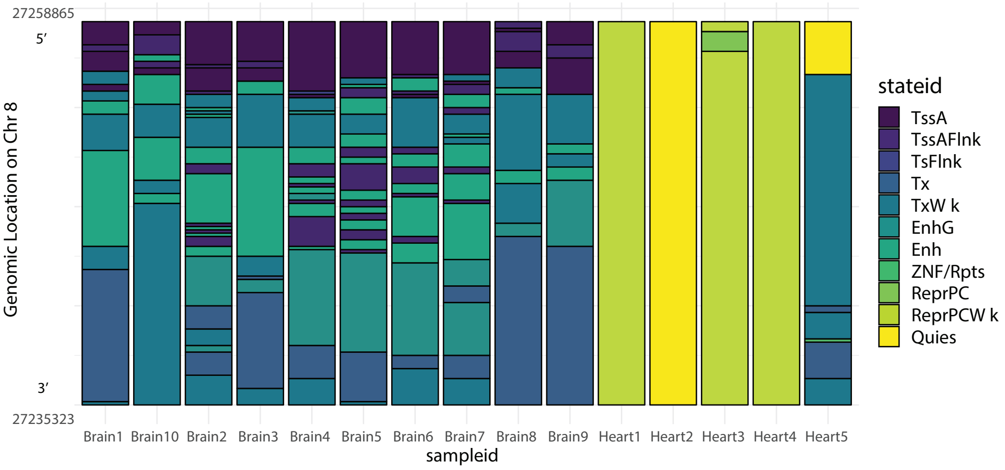


Figure 3.4: Chromatin state sequences of gene *STMN4* in all the 10 brain samples and the 5 heart samples. Different chromatin states are represented by different colors. The y-axis indicates the genomic locations of various chromatin states across these 15 samples.

	DESeq2	edgeR
Total number of genes	17784	17784
Number of DE genes ($p < 0.05$)	5906	6251
DE genes in top 200 by EpiAlign	143	146
p-value of hyper-geometric test	$< 10^{-30}$	$< 10^{-30}$

Table 3.2: Comparison of the 200 genes with differential chromatin state sequences identified by EpiAlign and the differentially expressed (DE) genes identified by DESeq2 or EdgeR. DESeq2 and edgeR identify 5906 and 6251 DE genes between all 3 brain samples and all 4 heart samples from the 17,784 protein-coding genes in the Roadmap RNA-seq datasets. A hypergeometric test is used to check the significance of the enrichment of the top 200 genes identified by EpiAlign in the two sets of DE genes. The two resulting p-values are both significant.

tissue-specific epigenomic regions.

3.3.2 Horizontal Alignment: Analysis of Frequent Chromatin State Sequence Patterns within an Epigenome

Motivated by the fact that similar chromatin state sequences may encode similar biological functions, here we use EpiAlign to analyze frequent chromatin state sequence patterns within an epigenome. We introduce the "**horizontal alignment**," which takes the chromatin state sequence of a region as the query and searches for its best hit except itself within an epigenome. We first divide a given epigenome into regions of 500 kb length, and then we align the chromatin state sequence of each region (i.e., the "query") to those of other regions to find the best match. It is worth noting that the alignment scores of multiple query chromatin state sequences are not directly comparable. To normalize the alignment scores, we align every query chromatin state sequence to randomized chromatin state sequences, which serve as a negative control (see Supplementary section 3 for details). Then for every region, we define the normalized alignment score of its best hit except itself (when the region is used as the query) as its **horizontal alignment score**. A high score indicates that the region shares a highly similar and non-random chromatin state sequence with another region in the same epigenome, implying that the region's chromatin state sequence pattern is likely biologically meaningful.

With horizontal alignment scores, we can represent every epigenome by a vector, whose length is the number of regions and whose entries are the regions' horizontal alignment scores. As mentioned above, horizontal alignment scores measure whether their corresponding regions contain biologically meaningful chromatin state patterns, which are expected to be largely consistent across epigenomes of the same tissue. We use the Roadmap samples to calculate the horizontal alignment scores for all regions in all epigenomes. Then we represent every epigenome by a horizontal alignment score vector. To verify the biological meaning of the vector representation, we calculate the pairwise Pearson correlations between epigenomes and perform an average-linkage hierarchical clustering of epigenomes based on the (1−Pearson correlation) distance metric. The clustering result matches our expectation: samples from the same tissue are clustered together, confirming that the horizontal alignment scores are indeed consistent across the samples from the same tissue (Figure 3.5).

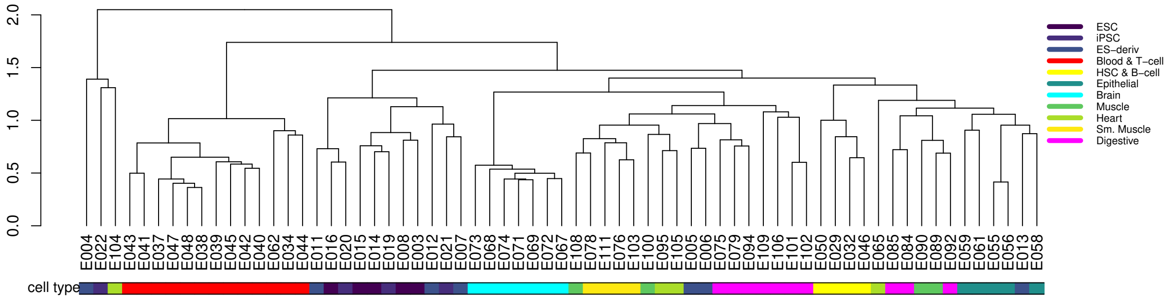


Figure 3.5: Clustering based on the correlation matrix of horizontal alignment scores of Roadmap epigenomes. Samples from the same tissue or cell type are clustered together, indicating that horizontal alignment scores are highly correlated between samples from the same tissue or cell type.

3.3.2.1 EpiAlign distinguishs real epigenomes from randomized ones

We further perform a simulation study to technically validate the efficacy of EpiAlign in terms of horizontal alignment. Our goal is to check if EpiAlign is able to distinguish real epigenomes from randomized epigenomes, which serve as a negative control. We calculate horizontal alignment scores using EpiAlign on all the 127 Roadmap samples based on the 15-state ChromHMM annotation. In addition to each real epigenome, we also generate a randomized epigenome and two hybrid epigenomes for comparison. Here the randomized epigenome is generated in the same way as in the normalization step for calculating horizontal alignment scores (see Supplementary section 3 for details). To contrast real and randomized epigenomes, we also generate a hybrid epigenome as a semi-negative control by mixing the real and randomized epigenomes of every chromosome, so that a hybrid epigenome is composed of alternating real regions and randomized regions. (see Supplementary section 4 for details)

We use an ESC (embryonic stem cell) sample (Roadmap ID E003) as an example and calculate horizontal alignment scores in four epigenomes: the real ESC epigenome, a randomized epigenome, and two hybrid epigenomes. We summarize the distributions of horizontal alignment scores in the real and randomized epigenomes in Figure 3.2(a). As expected, the regions in the real epigenome have an average alignment score higher than 0, while the average score of regions in the randomized epigenome is close to 0. For each of these four epigenomes, we find the top 500 non-overlapping regions with the highest horizontal alignment scores. As expected, the top regions in the real epigenome have scores significantly higher than those in the randomized and hybrid epigenomes (Figure 3.2(b)), an observation consistent with the fact that a high score indicates a region likely to have a biologically meaningful chromatin state pattern. Moreover, for hybrid epigenomes, almost all the top 500 regions are those generated from the real epigenome (Figure 3.2(c)), again confirming that real chromatin state patterns are more biologically meaningful than randomized patterns. Overall, our results suggest that EpiAlign can powerfully distinguish real biological epigenomes from randomized epigenomes.

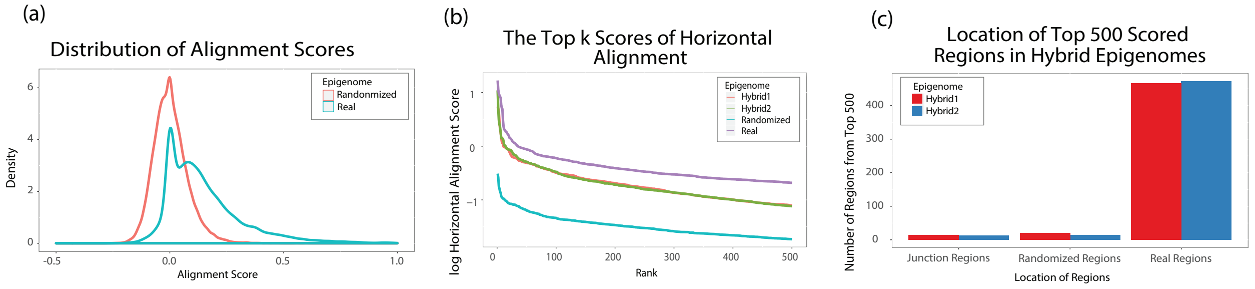


Figure 3.6: Horizontal alignment results of embryonic stem cell sample E003. (a) The distribution of horizontal alignment scores of regions in real and randomized epigenomes. (b)The top 500 highest horizontal alignment scores ($\log_{10}$ transformed) in real, randomized and hybrid epigenomes. Scores in the real epigenome are always the highest given the same rank. (c) Locations of the regions with the top 500 horizontal alignment scores in the two hybrid epigenomes. The three panels together indicate that the real epigenome contains non-random chromatin state sequential patterns captured by EpiAlign.

3.3.2.2 Comparison of EpiAlign with alternatives

We further validate our EpiAlign algorithm with equal weights by comparing it with two alternative approaches. The first is a variant of EpiAlign using frequency-based weights, which are determined by the frequencies of chromatin states (see Supplementary section 1 for details). The second is a naïve alignment method, in which we first calculate the proportion of each chromatin state in two regions (chromatin state sequences) to obtain two proportion vectors $P_1 = (p_{11}, p_{12}, \dots, p_{1Q})^\intercal$ and $P_2 = (p_{21}, p_{22}, \dots, p_{2Q})^\intercal$, where Q is the number of unique chromatin states in the annotation(e.g., $Q = 15$ in this case). The naïve alignment score is a similarity measure defined as $M^{\text{naïve}} = -||P_1 - P_2||_2^2 = -\sum_{i=1}^{Q}(p_{1i} - p_{2i})^2$. The naïve method directly compares two chromatin state sequences based on their state proportions, and it does not use a dynamic programming approach as does in EpiAlign. However, given that similar chromatin state sequences share similar frequency vectors, the naïve method is also a biologically meaningful approach.

Note that EpiAlign (with equal weights), the frequency-based variant of EpiAlign, and the naïve method do not have horizontal alignment scores on the same scale and cannot be compared directly, so we compare the three approaches by evaluating the biological meaning of the regions they find with high scores. Since gene regions are expected to share some common chromatin state patterns (i.e., promoter, transcription start site, transcribed region, and transcription ending site), a good alignment method is expected to assign high horizontal alignment scores to gene regions. In other words, genes expressed in a tissue are expected to have high horizontal alignment scores in the tissue's epigenome. Hence, we design two evaluation criteria: one is the enrichment of known tissue-associated genes, i.e., the non-house-keeping genes highly expressed in a tissue [94], in regions with high alignment scores; the other criterion is the enrichment of annotated genes. The greater the enrichment, the better the alignment method. We apply each of the three approaches to do horizontal alignment and check the overlap between tissue-associated genes or annotated genes and each approach's top-aligned regions, which receive the highest horizontal alignment scores. We perform this evaluation on 16 samples: 5 ESC, 4 heart and 7 brain samples. For each sample, we collect the top 500 regions with the highest alignment scores found by each approach and count the numbers of tissue-associated genes from Yang et al. [94] and annotated genes from Kent et al. [95] that overlap with these regions. From the results shown in Figure 3.7, we see that EpiAlign outperforms the naïve method in detecting annotated genes and tissue-associated genes. In addition, we observe that the frequency-based weights do not have apparent advantages over the equal weights, suggesting that we may use EpiAlign with equal weights as the default.

3.3.2.3 Motif Analysis

As a further investigation, we check if the regions with top horizontal alignment scores share any chromatin state patterns in common. We apply EpiAlign to perform horizontal alignment within the epigenome of the embryonic stem cell sample E003, and we select the top 200 regions with the highest horizontal alignment scores. To investigate whether common chromatin state patterns exist among these regions, we calculate the pairwise alignment scores between each pair of these top 200 regions. We normalize the pairwise alignment scores and store them in a 200×200 symmetric matrix $\mathbf{A}$, whose (i, j)-th entry A_{ij} represents the normalized alignment score of regions i and j and is defined as

$$A_{ij} = \begin{cases} 1 & \text{if } i = j \\ \frac{\text{alignment score of regions } i \text{ and } j}{\alpha(\max_{k \neq r} \text{alignment score of regions } k \text{ and } r)} & \text{otherwise} \end{cases}, \tag{3.8}$$

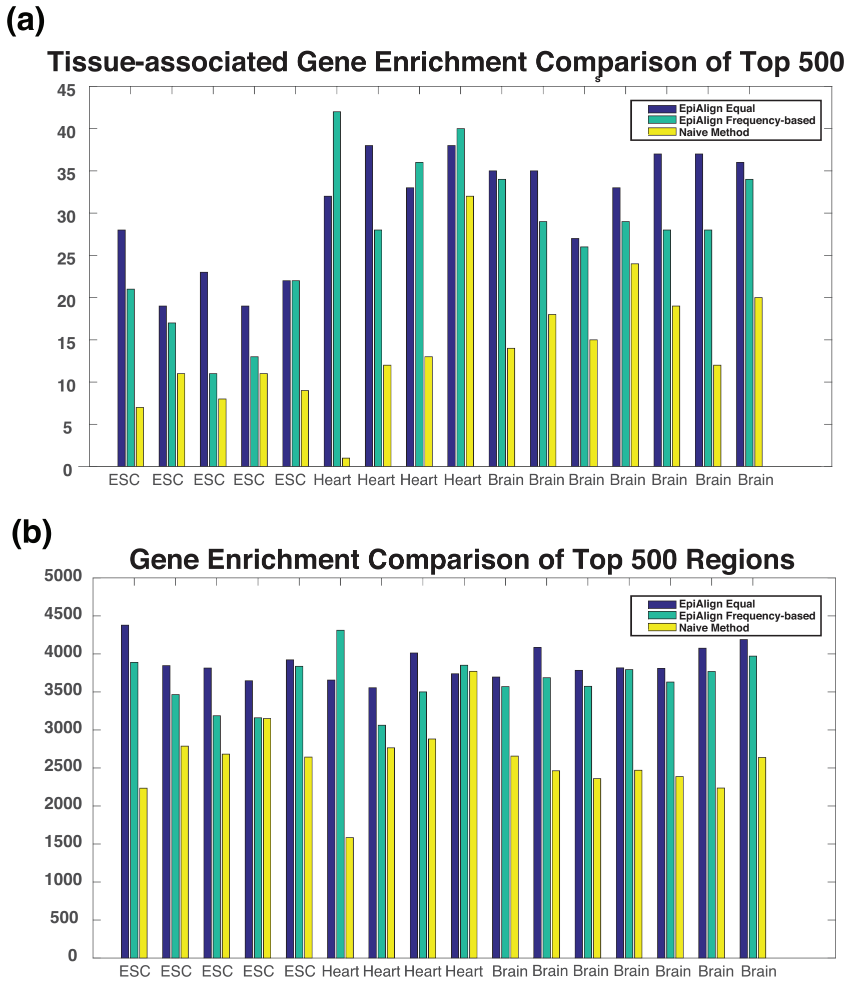


Figure 3.7: Comparison of EpiAlign, EpiAlign with frequency-based weights, and the naïve method using 16 Roadmap samples (5 ESC, 4 heart, and 7 brain samples from the 92 samples with 18-state ChromHMM annotation). (a) The number of tissue-associated genes that overlap with the top 500 regions with the highest horizontal alignment scores found by each approach. (b) The number of annotated genes that overlap with the same three sets of top 500 regions.

where $\alpha = 1.1$ ensures that $0 < A_{ij} < 1$ for all $i \neq j$. We then define a distance matrix $\mathbf{D}$, whose (i, j)-th entry is $D_{ij} = 1 - A_{ij}$. We then perform hierarchical clustering with average linkage on the top 200 regions based on $\mathbf{D}$, and we display the clustering result in Figure 3.8.

From the heatmap in Figure 3.8, we see that the top 200 regions are well partitioned into four clusters, indicating that regions in the same cluster share similar chromatin state patterns. (Supplementary Table 3) We inspect each of these four clusters to identify its representative chromatin state patterns, which we refer to as *motifs* in the following text. For notation simplicity, we use alphabets “a” to “o” to denote chromatin states 1 to 15.

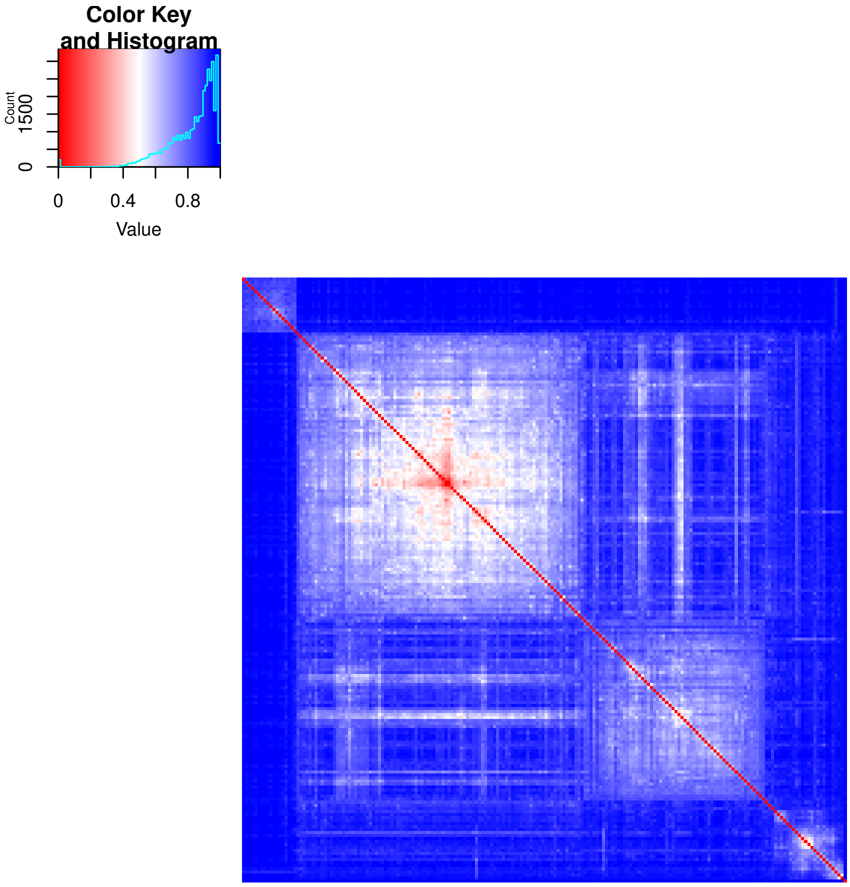


Figure 3.8: Heatmap of pairwise distances of the top 200 regions, identified by the horizontal alignment on embryonic stem cell sample E003. Based on the distance matrix **D**, the top 200 regions are grouped into 4 clusters by average-linkage hierarchical clustering.

Using the motif-discovery tool MEME [96], we find that all the four clusters are characterized by certain motifs. As annotated by the 15-state ChromHMM model [97], the state "o" denotes the quiescent state and lacks a good biological interpretation, so we only consider the motifs without "o". We find that cluster 1 is characterized by the "ihih"-repeat motif; cluster 2 is characterized by the "egeg"-repeat motif; cluster 3 is characterized by "eded" motif; cluster 4 is characterized by the "egeg" motif and "mlml" motif. Based on the ChromHMM annotation, the state "i" represents heterochromatin, while "h" represents ZNF genes and repeats. Since existing evidence shows that human heterochromatin proteins form large domains containing KRAB-ZNF genes [98], the "ihih"-repeat motif may represent functional non-coding regions. Since "d" denotes strong transcription, "e" denotes weak transcription, and “g” denotes enhancer, the “egeg”-repeat motif may be an evidence of transcriptional enhancers [99] and the “eded”-repeat motif may denotes transcriptional regions. In the “mlml”-repeat motif, “m” and “l” represent repressed polycomb and bivalent enhancer, respectively. Since polycomb-repressed genes have permissive enhancers that initiate reprogramming [100], the “mlml”-repeat motif may be an indicator of polycomb-repressed gene regions. All these results show that the motifs discovered from the frequent chromatin state patterns are biologically meaningful and EpiAlign can help identify common chromatin state patterns in epigenomes of specific biological conditions.

3.3.2.4 Cross-species application of EpiAlign

We further investigate the application of EpiAlign to comparing human and mouse chromtain state sequences. We use the epigenetic data from Yue et al. (2014), where mouse and human samples were used together to train a 7-state ChromHMM model[101]. We investigate two liver samples, one from human and one from mouse. As homologous genes are expected to exhibit more similar functions than non-homologous genes[102], we expect to observe larger alignment scores between chromatin state sequences of homologous genes than those of non-homologous genes of similar sequence lengths. Our analysis is as follows. We first obtain mouse-human homologous gene pairs from Ensembl BioMart (Release 95) [103]. We sort the mouse genes with lengths 200-400 kb by gene lengths and divide the homologous gene pairs into 12 groups each with 50 pairs, so that the mouse genes within a group have similar lengths. Within each group, we apply EpiAlign to each mouse-human homolog pair and each non-homolog pair. The results show that among the 12 groups, on average 16% the human genes have the highest chromatin state sequence alignment scores with their corresponding mouse homologs, suggesting that homologous genes tend to share similar epigenetic patterns. We also look at the GO terms of the homolog pairs that have the highest alignment scores in each group. The result (see Supplementary Table S4) shows that homologous genes with high alignment scores are also very similar in molecule functions and biological processes. The result also indicates that EpiAlign can identify homologous genes whose epigenetic patterns are more conserved in evolution, shedding new insights into translating scientific discoveries in mice into humans.

3.4 Discussion

In this article, we propose the EpiAlign algorithm for alignment of chromatin state sequences learned from multi-track epigenomic signals. We demonstrate that EpiAlign can be a powerful tool for studying the epigenetic dynamics along the same epigenome or across multiple epigenomes, based on both simulation and real data studies.

First, our current alignment results are based on ChromHMM, which learns and characterizes from multi-track epigenomic signals. There are also other tools for pattern discovery in chromatin structures, such as Segway [15], which constructs a dynamic Bayesian network instead of HMM, EpiCSeg [85], which uses natural numbers instead of binarized signals as used by ChromHMM, and IDEAS [87], which jointly characterizes epigenetic dynamics across multiple human cell types. It would be interesting to compare these tools with ChromHMM to analyze how the chromatin state annotation affects the alignment results of EpiAlign. If the output results of ChromHMM or other segmentation tools can be filtered or improved based on additional biological experiments, this can also help EpiAlign obtain more accurate and robust results. Besides, we find likely noisy ChromHMM annotations that need further biological validation (see Supplementary section 12). To account for such possible inaccuracy in chromatin state sequences, we may improve EpiAlign by incorporating the posterior probabilities of chromatin states output by ChromHMM into the calculation of alignment scores. Moreover, ChromHMM is an unsupervised algorithm that requires a pre-specified number of states; thus, its chromatin state labels may not be fully biologically meaningful. For example, some genomic regions would be assigned to different chromatin states given different numbers of states. This leads to additional noise in ChromHMM annotations. To account for such noise, we may correct chromatin state labels by using the sequential information in neighboring states.

Second, in the EpiAlign algorithm, an important step before alignment is the compression of the chromatin state sequences. Chromatin states of different regulatory functions can vary greatly in their lengths [104], but the length information itself is not always informative of the change of epigenetic marks along the genome. Specifically, the quiescent/low state often appear in extremely long stretches, whose lengths are not useful for comparing chromatin state sequences (see Supplementary Figure S1). Therefore, we add a compression step to capture and extract the dynamics of chromatin states among biological samples. We have also tested the pre-compression alignment algorithm, but it is not able to distinguish the randomized chromosome from the real one, suggesting that compression is necessary for detecting biologically meaningful chromatin state patterns. However, we realize that this compression step still has room for improvement. For example, several previous studies have shown that broad/sharp H3K4me3 domains have distinct functions [105–107], implying that the length information of certain chromatin states is important for vertical alignment that compares a region across samples. Future refinement of the compression step, or refinement of length information usage after compression, should consider multiple aspects: a chromatin state's confidence (whether it is likely noisy) and importance (whether its length information is informative), as well as the analysis needs (vertical or horizontal alignment), among others.

Third, EpiAlign is essentially an unsupervised algorithm, but the flexibility of the weight function allows EpiAlign to incorporate prior knowledge into the alignment procedure by assigning different weights to different chromatin states. For example, the frequency-based weights lead the algorithm to favor the alignment of less frequent patterns compared to background patterns, which frequently exist along the epigenome. In practical applications, one may adjust the weight function to reflect the important elements in specific problems. For instance, the weight can incorporate the transcription start sites (TSSs) in genome annotation when transcriptional regulation is of particular importance.

Fourth, EpiAlign depends on two tuning parameters: ϵ_N and ϵ_D for penalizing mismatches and gaps in the alignment. Similar parameters are also necessary for classical alignment algorithms designed for DNA and protein sequences such as BLAST. For example, the ϵ_D in EpiAlign is analogous to the Gap Extend Penalty in BLAST. The NCBI BLAST, an online tool that implements the BLAST algorithm, sets the Gap Extend Penalty to 1 by default. In EpiAlign, we also set ϵ_D to 1 by default. In BLAST, a substitution matrix is used to score matches/mismatches, and multiple substitution matrices have been constructed for users to select based on alignment purposes. In EpiAlign, we set ϵ_N to 1.5, which is equivalent to a substitution matrix with diagonal entries as 1 and off-diagonal entries as -1.5. Given that the alignment of epigenetic sequences is new to this field, how to construct more specialized substitution matrices for chromatin states is an important future research question.

Finally, in some computationally efficient sequence alignment algorithms, hash tables or tree-based data structures are utilized to index the database, and these techniques have greatly increased the efficiency of query retrieval. EpiAlign can also benefit from similar techniques and further improve its computation efficiency.

Two other computational methods, EpiCompare [108] and ChromDiff [109], have been developed to compare chromatin states between samples. They test for the difference of a single chromatin state's frequency in a genomic region between two groups of samples. EpiCompare restricts the region of interest to a 200 bp window, which corresponds to a single chromatin state output by ChromHMM. A useful functionality of EpiCompare is that it searches for the 200 bp windows where the specified chromatin state is enriched only under one condition. Compared with EpiCompare, ChromDiff is more flexible and allows the region to have any length greater than 200 bp. Another advantage of ChromDiff is that it normalizes the chromatin state frequencies to reduce the effects of confounding covariates. A common limitation of ChromDiff and EpiCompare is that they can only compare chromatin state frequencies between two conditions in the same genomic region, and they require multiple samples under each condition. In contrast, EpiAlign can perform pairwise alignment between any two chromatin sequences, either coming from the same genomic region in two samples or two different genomic regions in one sample. In other words, EpiAlign does not pose any restrictions on the choice of genomic regions or the sample size. Furthermore, EpiAlign has two unique advantages. First, it simultaneously uses the sequential information encoded in multiple chromatin states. Second, it outputs an alignment score that integrates this sequential information. Hence, EpiAlign enables horizontal alignment and query search, allowing us to extract chromatin state patterns that carry tissue-associated characteristics. These patterns are shown to be biologically meaningful in our motif analysis and have a strong capability in grouping epigenomic samples of the same cell type in horizontal alignment.

In terms of biological applications, the biggest strength of EpiAlign is its ability to identify common chromatin state patterns and how they are conserved or divergent between cell types. This strength will pave the way for identifying regulatory domains defined by combinatorial effects of strings of cis-elements. Specifically, the vertical analysis based on EpiAlign will reveal tissue-specific genes and regulatory regions that share common chromatin state patterns within a tissue type, and such patterns will serve as the basis of defining new regulatory domains. We have also demonstrated that EpiAlign has found meaningful chromatin state motifs. Besides, EpiAlign is able to distinguish tissue-associated genes. These results suggest the potential of EpiAlign as a useful bioinformatic tool to discover tissue-specific gene regulation. Moreover, the alignment scores calculated by EpiAlign can serve as a covariate when constructing functional genomic networks, thus allowing the network to incorporate similarities of chromatin structures as a factor. Further, EpiAlign applies to 3D genomic analysis to address the question if there are chromatin state patterns in regions with a specific 3D structure such as a loop.

3.5 Acknowledgments

We deeply appreciate the insightful feedbacks from Prof. Jason Ernst at UCLA and Dr. Yucheng T. Yang at Yale University. We also thank Prof. Changshui Zhang at Tsinghua University for supporting H.Z.'s research.

3.6 Supplementary Material

S3.6.1 Frequency-based weights

The frequency-based weight of the k-th state in chromatin state sequence S is $W^{Fb,w}(k,S)$, which is defined as:

$$W^{Fb,w}(k,S) = GS^w(k,S) \cdot RS^w(k,S) \cdot LS(k,S)$$

where w is a window size parameter, $GS^w(k,S)$ is the two-state frequency score, $RS^w(k,S)$ is the one-state frequency score, and $LS(k,S)$ is the length score.

In the subsequence $S^{[k-w,k+w]}$, if neither of pattern $S^{[k-1,k]}$ or $S^{[k,k+1]}$ occur elsewhere in the subsequence, $GS^w(k,S) = 1$; if both patterns occur somewhere in the subsequence, $GS^w(k,S) = 4$; otherwise $GS^w(k,S) = 2$.

The one-state frequency score $RS^w(k,S)$ represents the frequency of each state and gives more frequent states smaller scores. For $S^{[k-w,k+w]}$, we rank the chromatin states in this window by their frequencies, from the highest to the lowest. Then

$$RS^w(k,S) = 1 + \frac{\text{rank}(S^{[k]}) - 1}{\text{number of unique states in } S^{[k-w,k+w]} - 1} \in [1,2]\,.$$

In the compression process of EpiAlign, we compress consecutive occurrences of the same state into a state label. We also obtain an occurrence number for each state. For example, a chromatin state sequence `abbcc` is represented by a compressed state sequence $S =$ `abc` and a state occurrence sequence $L =$ `122`. The length score $LS(k,S)$ is based on the occurance sequence L. If $L^{[k]} = 1$, $LS(k,S) = 0.5$; otherwise $LS(k,S) = 1$.

S3.6.2 Average length of stretches of the same chromatin state

The motivation for doing compression before alignment comes from the fact that most uncompressed sequences contain long stretches of the same chromatin state. Here we use letter “a” to “o” to denote chromatin state 1 to 15 from the Roadmap 15-state annotation. We calculate the average length of consecutive same chromatin state for each state. From the results in Figure S1, we can see that the chromatin state “o”, which means quiescent/low state, is much longer than other states before compression. As the length information of such a state is hardly biologically meaningful. The compression step is needed for addressing this issue by turning the focus onto the more biologically meaningful chromatin state patterns.

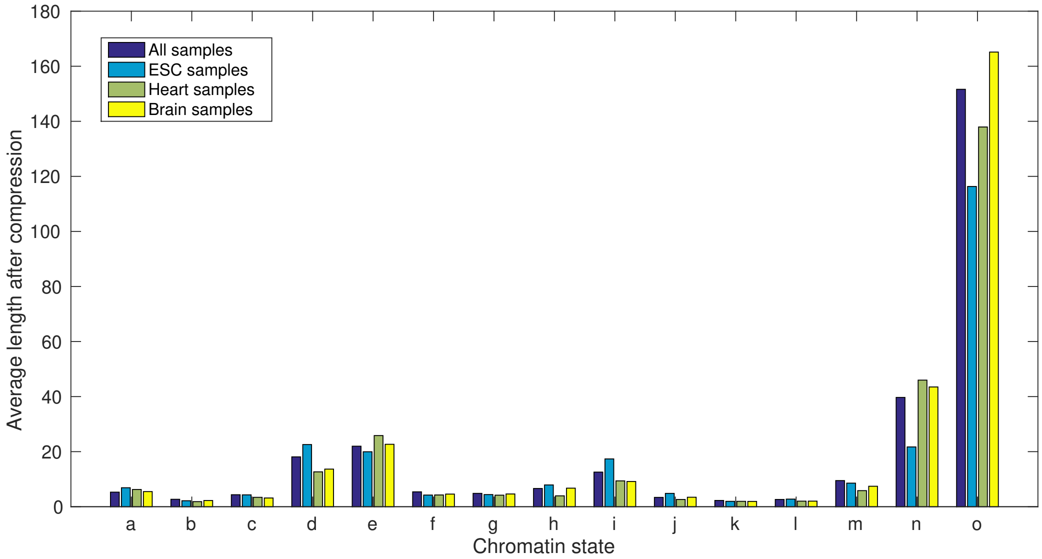


Figure 3.9: Average lengths of stretches of a single chromatin state in ESC, heart and brain samples. Letters ’a’ to ’o’ refer to chromatin states 1 to 15 in the Roadmap 15-state chromHMM annotation. For each state, we calculated the average length of its stretches, i.e., consecutive occurrences. It is obvious that the chromatin state ’o’, i.e., the quiescent/low state, has much longer stretches than other states do.

Horizontal alignment scores

It is worth noting that the alignment scores of multiple query chromatin state sequences are not directly comparable. To normalize the alignment scores, we align every query chromatin state sequence to randomized chromatin state sequences, which serve as a negative control. For each region in the real epigenome, this region is used as the “query” and aligned to each of the three randomized epigenomes to obtain its hit in that randomized epigenome. Here the randomized epigenomes have the same lengths as their real counterparts and are generated by the Markov rule, with a state transition probability matrix per chromosome based on the real epigenome. The alignment scores of the three hits are then averaged as the baseline score of this query region. We use Q_i to denote the alignment score of region

i's hit in the real epigenome, P_i to denote the baseline score of region i, and define the horizontal alignment score of region i as $\frac{Q_i - Pi}{Pi}$. A high score indicates that the region shares a highly similar and non-random chromatin state sequence with another region in the same epigenome, implying that the region's chromatin state sequence pattern is likely biologically meaningful.

S3.6.3 Generation of hybrid epigenomes

For every chromosome, we first divide both its real chromatin state sequence ("epigenome") and their randomized counterparts into non-overlapping regions of 50 million bp length. Then for the i-th region in the hybrid epigenome, its chromatin state sequence is set as the sequence of the i-th region in the real epigenome when i is even, or as the sequence of the i-th region in the randomized epigenome when i is odd. We can easily generate another hybrid epigenome if we switch the odd and even regions.

Figure S2 shows that when we use the chromatin state sequences of gene regions as queries, the best hits (regions that have the highest horizontal alignment scores with the query) reported by EpiAlign are very similar to the query in terms of chromatin state patterns.

S3.6.4 Comparison of uncompressed sequences and compressed sequences in Vertical Alignment

We also use the vertical alignment to justify our choice of aligning compressed chromatin state sequences instead of original uncompressed sequences. We repeated the vertical alignment analysis on all brain-specific genes and all heart-specific genes among the brain and heart samples, using the uncompressed sequences instead of the compressed chromatin state sequences. Then we performed the same two-sample one-sided Wilcoxon test between brain-vs-brain alignment scores and brain-vs-heart alignment scores on these selected genes, and we denote the resulting p-values as uncompressed p-values. Next we compare these uncompressed p-values with their corresponding p-values we obtain previously based on the compressed sequences. We counted the number of significantly different genes, which have a p-value less than 0.05 after Bonferroni correction, from both analyses. From compressed sequences, 112 out of 327 tissue-specific genes are significant and from uncompressed 120 out of 327 are significant. We also conducted two-sample Wilcoxon test between the original p-values and p-values from uncompressed sequences. The result shows no significant difference in distribution (p-value = 0.509). These results show that the resulting p-values from the compressed sequences are very similar to those from the uncompressed sequences. Considering that alignment of uncompressed sequences is much more time-consuming (takes 20 times more time than compressed sequences), the compression step makes the alignment algorithm more effective.

S3.6.5 Examples of vertical alignment on tissue-associated genes

Since epigenetic marks carry important regulatory information relevant to cell differentiation, chromatin states learned from these marks should also contain cell-type characteristic patterns. For a tissue-associated gene, we should expect to observe significantly higher similarity of chromatin states within its associated cell type than the counterpart similarity between cells of other cell types. We implement vertical alignment on the Roadmap dataset on some tissue-associated genes [1]. Taken an ESC-associated gene *ANAPC1* as an example, we use the alignment scores calculated by EpiAlign to compare the similarity of *ANAPC1*'s chromatin state sequences in different cell types. We first extract the chromatin state sequence of *ANAPC1*'s chromatin region from in each epigenome. Then, we use EpiAlign to calculate the alignment scores of these chromatin state sequences between each pair of the 127 epigenomes, resulting in $8,001$ pairwise alignment scores in total. We consider these $8,001$ alignment scores as the population and refer to the alignments scores between epigenomes of the same cell type (i.e., ESC) as the Group A, and the whole population as the Group B. As there are 8 ESC epigenomes, we obtain 28 alignment scores in the Group A and then calculate the percentile of each of these 28 scores in the population. As shown in Figure S3, 11 out of the 28 scores are among the upper 5% percentile, and the average percentile of alignment scores in group A is 0.256. We also perform a one-tailed t-test to compare the alignment scores of the two groups. The p-value of the test is 0.001, which suggests that the chromatin states corresponding to the ESC-associated gene *ANAPC1* have more similar patterns among ESC samples in comparison with the other cell types in the Roadmap dataset. We also use the alignment scores based on the naïve method and repeat all the analysis above.

From Figure S3, we can see that compared to the naïve method, EpiAlign can better distinguish alignment scores among ESCs from others. Similar results are observed for brain and heart too. These results indicate that for a specific cell type, EpiAlign is able to detect the similarity of chromatin states of its tissue-associated genes, suggesting that EpiAlign may be used to differentiate a given cell type from the other tissue and cell types, by evaluating the similarity of epigenetic signals on its associated genes. Also, these results show that EpiAlign can be used to identify tissue-associated regions.

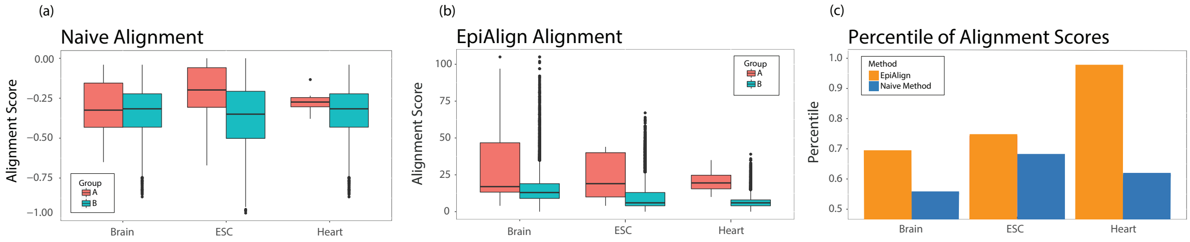


Figure 3.10: (a)-(b) Boxplots of pairwise alignment scores of chromatin state sequences of a tissue-associated gene within same the cell type (Group A) and across all samples (Group B). We choose an ESC-associated gene *ANAPC1*, a brain-associated gene *AK5*, and a heart-associated gene *ACTN2*. (a) shows the alignment scores by the naïve method, and (b) shows the EpiAlign alignment scores. (c) shows the average percentile of group A scores in Group B for each alignment method.

We also perform hierarchical clustering of the 127 cells using chromatin state sequences of multiple tissue-associated genes. For example, for each of the 118 ESC-associated genes, we use EpiAlign to calculate all the pairwise alignment scores and form a 127×127 score matrix. We then normalize the scores by dividing the maximum so that for each gene i, we get a 127×127 normalized comparison matrix M^i. Then the final distance matrix D is calculated as $D_{jk} = -\sqrt{\sum_{i=1}^{118} M_{jk}^i}$. Finally, we perform complete-linkage hierarchical clustering on the 127 epigenomes based on the distance matrix. The heatmap of the distance matrix and the clustering results are shown in Figure S4(b). The heatmap can roughly distinguish ESC samples from the other cell types. In hierarchical clustering, 7 out of 8 ESC samples are successfully grouped together when the cluster number is set as 10. Similarly, we perform the above analysis based on brain-associated genes, and the heatmap is shown in 3.3(a). The brain samples can be clearly differentiated. In addition, all the 10 brain samples are successfully grouped together by hierarchical clustering when cluster number is set as 10. The above results confirm EpiAlign's capability to search for similar chromatin state patterns and suggest that chromatin states of the same genomic region are more similar within cell types.

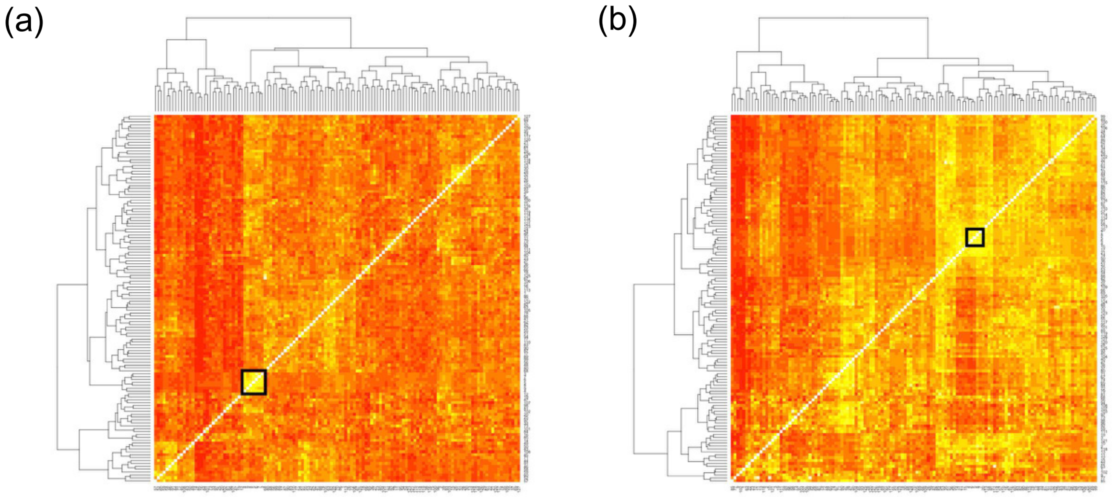


Figure 3.11: Clustering results using (a) brain-associated genes or (b) ESC-associated genes. Samples in black boxes are (a) brain samples and (b) ESC samples.

S3.6.6 Male-vs-female vertical alignment

	Gene.stable.ID	Gene.name	Chromosome	Gene.description
1	ENSG00000012660	ELOVL5	6	ELOVL fatty acid elongase 5
2	ENSG00000111832	RWDD1	6	RWD domain containing 1
3	ENSG00000112232	KHDRBS2	6	KH RNA binding domain containing, signal transduction associated 2
4	ENSG00000135968	GCC2	2	GRIP and coiled-coil domain containing 2
5	ENSG00000136485	DCAF7	17	DDB1 and CUL4 associated factor 7
6	ENSG00000138035	PNPT1	2	polyribonucleotide nucleotidyltransferase 1
7	ENSG00000138398	PPIG	2	peptidylprolyl isomerase G
8	ENSG00000139053	PDE6H	12	phosphodiesterase 6H
9	ENSG00000168958	MFF	2	mitochondrial fission factor
10	ENSG00000170293	CMTM8	3	CKLF like MARVEL transmembrane domain containing 2
11	ENSG00000173572	NLRP13	19	NLR family pyrin domain containing 13
12	ENSG00000197360	ZNF98	19	zinc finger protein 98

Table S3.3: Genes not on chromosome X among the top 200 genes with the smallest p-values from comparing male-vs-male scores and male-vs-female scores.

S3.6.7 Brain-vs-heart vertical alignment

Genes with the smallest p-values from one-sided Wilcoxon test

We perform the two-sample one-sided Wilcoxon test between the brain-vs-brain alignment scores and the brain-vs-heart alignment scores for all the protein-coding genes. The top 200 genes that we use to perform the gene ontology enrichment analysis are listed in Table S2.

	Gene.stable.ID	Chromosome	Gene.name	Strand	Gene.start..bp.	Gene.end..bp.
1	ENSG00000004700	12	RECQL	-1	21468911	21501669
2	ENSG00000006047	17	YBX2	-1	7288252	7294615
3	ENSG00000015592	8	STMN4	-1	27235323	27258420
4	ENSG00000033122	1	LRRC7	1	69568398	70151945
5	ENSG00000034053	15	APBA2	1	28884483	29118315
6	ENSG00000047365	4	ARAP2	-1	35948221	36244509
7	ENSG00000050438	12	SLC4A8	1	51391317	51515763
8	ENSG00000054282	1	SDCCAG8	1	243256034	243500092
9	ENSG00000056487	22	PHF21B	-1	44881162	45009999
10	ENSG00000064270	16	ATP2C2	1	84368527	84464187

11	ENSG00000065609	6	SNAP91	-1	83552880	83709691
12	ENSG00000066468	10	FGFR2	-1	121478334	121598458
13	ENSG00000067221	15	STOML1	-1	73978923	73994622
14	ENSG00000067798	12	NAV3	1	77324641	78213008
15	ENSG00000070501	8	POLB	1	42338454	42371808
16	ENSG00000073803	3	MAP3K13	1	185282941	185489097
17	ENSG00000074211	4	PPP2R2C	-1	6320578	6563600
18	ENSG00000077009	19	NMRK2	1	3933103	3942416
19	ENSG00000078295	5	ADCY2	1	7396208	7830081
20	ENSG00000078725	9	BRINP1	-1	119153458	119369467
21	ENSG00000084628	1	NKAIN1	-1	31179745	31239554
22	ENSG00000088766	20	CRLS1	1	6006090	6040053
23	ENSG00000089225	12	TBX5	-1	114353931	114408442
24	ENSG00000091129	7	NRCAM	-1	108147623	108456717
25	ENSG00000095397	9	WHRN	-1	114402080	114505450
26	ENSG00000100290	22	BIK	1	43110748	43129712
27	ENSG00000100433	14	KCNK10	-1	88180103	88326907
28	ENSG00000100505	14	TRIM9	-1	50975262	51096061
29	ENSG00000102383	X	ZDHHC15	-1	75368427	75523502
30	ENSG00000104112	15	SCG3	1	51681353	51721031
31	ENSG00000104833	19	TUBB4A	-1	6494319	6502848
32	ENSG00000105048	19	TNNT1	-1	55132794	55149354
33	ENSG00000106780	9	MEGF9	-1	120600813	120714470
34	ENSG00000107438	10	PDLIM1	-1	95237572	95291024
35	ENSG00000108001	10	EBF3	-1	129835283	129963841
36	ENSG00000108187	10	PBLD	-1	68282660	68333049
37	ENSG00000108688	17	CCL7	1	34270221	34272242
38	ENSG00000108830	17	RND2	1	43025241	43032036
39	ENSG00000109472	4	CPE	1	165361194	165498320

40	ENSG00000109654	4	TRIM2	1	153152342	153339320
41	ENSG00000109956	11	B3GAT1	-1	134378504	134411918
42	ENSG00000110042	11	DTX4	1	59171430	59208587
43	ENSG00000110076	11	NRXN2	-1	64606174	64723188
44	ENSG00000110628	11	SLC22A18	1	2899721	2925246
45	ENSG00000111605	12	CPSF6	1	69239537	69274358
46	ENSG00000111726	12	CMAS	1	22046174	22065674
47	ENSG00000112041	6	TULP1	-1	35497874	35512938
48	ENSG00000112139	6	MDGA1	-1	37630679	37699306
49	ENSG00000112290	6	WASF1	-1	110099819	110180004
50	ENSG00000112379	6	ARFGEF3	1	138161921	138344663
51	ENSG00000113456	5	RAD1	-1	34905264	34918989
52	ENSG00000113460	5	BRIX1	1	34915376	34925996
53	ENSG00000113645	5	WWC1	1	168291651	168472303
54	ENSG00000115041	2	KCNIP3	1	95297304	95386083
55	ENSG00000115239	2	ASB3	-1	53532672	53787610
56	ENSG00000117020	1	AKT3	-1	243488233	243851079
57	ENSG00000117595	1	IRF6	-1	209785623	209806175
58	ENSG00000118322	5	ATP10B	-1	160563120	160852214
59	ENSG00000120937	1	NPPB	-1	11857464	11858931
60	ENSG00000120963	8	ZNF706	-1	101177878	101206193
61	ENSG00000121058	17	COIL	-1	56938187	56961054
62	ENSG00000121743	13	GJA3	-1	20138255	20161049
63	ENSG00000121904	1	CSMD2	-1	33513999	34165842
64	ENSG00000123560	X	PLP1	1	103773718	103792619
65	ENSG00000124641	6	MED20	-1	41905354	41921139
66	ENSG00000127955	7	GNAI1	1	79768028	80226181
67	ENSG00000128524	7	ATP6V1F	1	128862826	128865844
68	ENSG00000129250	17	KIF1C	1	4997948	5028401

69	ENSG00000129991	19	TNNI3	-1	55151767	55157773
70	ENSG00000130176	19	CNN1	1	11538717	11550323
71	ENSG00000130226	7	DPP6	1	153887097	154894285
72	ENSG00000130475	19	FCHO1	1	17747718	17788568
73	ENSG00000131409	19	LRRC4B	-1	50516892	50568045
74	ENSG00000131437	5	KIF3A	-1	132692628	132737638
75	ENSG00000132549	8	VPS13B	1	99013266	99877580
76	ENSG00000133216	1	EPHB2	1	22710839	22921500
77	ENSG00000133958	14	UNC79	1	93333219	93707876
78	ENSG00000135069	9	PSAT1	1	78297143	78330093
79	ENSG00000135269	7	TES	1	116210493	116258783
80	ENSG00000135298	6	ADGRB3	1	68635367	69389511
81	ENSG00000136155	13	SCEL	1	77535674	77645263
82	ENSG00000136193	7	SCRN1	-1	29920103	29990289
83	ENSG00000136574	8	GATA4	1	11676959	11760002
84	ENSG00000137266	6	SLC22A23	-1	3268962	3457022
85	ENSG00000139364	12	TMEM132B	1	125186836	125662377
86	ENSG00000140937	16	CDH11	-1	64943753	65126112
87	ENSG00000141448	18	GATA6	1	22169443	22202528
88	ENSG00000141574	17	SECTM1	-1	82321024	82334074
89	ENSG00000141738	17	GRB7	1	39737927	39747291
90	ENSG00000142949	1	PTPRF	1	43525187	43623666
91	ENSG00000143951	2	WDPCP	-1	63121383	63827843
92	ENSG00000144369	2	FAM171B	1	186693971	186765965
93	ENSG00000144857	3	BOC	1	113211003	113287459
94	ENSG00000145284	4	SCD5	-1	82629539	82798857
95	ENSG00000145555	5	MYO10	-1	16661914	16936276
96	ENSG00000145794	5	MEGF10	1	127290831	127465737
97	ENSG00000146005	5	PSD2	1	139795821	139844466

98	ENSG00000146352	6	CLVS2	1	122995971	123072927
99	ENSG00000147488	8	ST18	-1	52110839	52460959
100	ENSG00000147724	8	FAM135B	-1	138130023	138496822
101	ENSG00000147799	8	ARHGAP39	-1	144529179	144605816
102	ENSG00000148123	9	PLPPR1	1	101028709	101325135
103	ENSG00000149571	11	KIRREL3	-1	126423359	127003460
104	ENSG00000149596	20	JPH2	-1	44111695	44187578
105	ENSG00000150477	18	KIAA1328	1	36829106	37232172
106	ENSG00000150625	4	GPM6A	-1	175632934	176002664
107	ENSG00000152578	11	GRIA4	1	105609994	105982092
108	ENSG00000154229	17	PRKCA	1	66302636	66810743
109	ENSG00000155886	9	SLC24A2	-1	19507452	19786928
110	ENSG00000156475	5	PPP2R2B	-1	146581146	147084784
111	ENSG00000157103	3	SLC6A1	1	10992186	11039249
112	ENSG00000157423	16	HYDIN	-1	70807378	71230722
113	ENSG00000157851	2	DPYSL5	1	26847747	26950351
114	ENSG00000158014	1	SLC30A2	-1	26037252	26046133
115	ENSG00000158615	1	PPP1R15B	-1	204403387	204411791
116	ENSG00000162706	1	CADM3	1	159171609	159203313
117	ENSG00000163449	2	TMEM169	1	216081866	216102783
118	ENSG00000164107	4	HAND2	-1	173524969	173530229
119	ENSG00000164163	4	ABCE1	1	145097932	145129179
120	ENSG00000164532	7	TBX20	-1	35202430	35254147
121	ENSG00000164542	7	KIAA0895	-1	36324221	36390125
122	ENSG00000165312	10	OTUD1	1	23439458	23442390
123	ENSG00000165527	14	ARF6	1	49893092	49897054
124	ENSG00000165548	14	TMEM63C	1	77116568	77259495
125	ENSG00000165566	13	AMER2	-1	25161684	25172288
126	ENSG00000166501	16	PRKCB	1	23835946	24220611

127	ENSG00000166831	15	RBPMS2	-1	64739892	64775587
128	ENSG00000166922	15	SCG5	1	32641676	32697098
129	ENSG00000167553	12	TUBA1C	1	49188736	49274603
130	ENSG00000168280	2	KIF5C	1	148875250	149026759
131	ENSG00000168495	8	POLR3D	1	22245104	22254600
132	ENSG00000168958	2	MFF	1	227325151	227357836
133	ENSG00000170091	5	NSG2	1	174045604	174243501
134	ENSG00000170185	4	USP38	1	143184917	143223830
135	ENSG00000171954	19	CYP4F22	1	15508493	15552317
136	ENSG00000172379	15	ARNT2	1	80404350	80597937
137	ENSG00000172461	6	FUT9	1	96015984	96215612
138	ENSG00000172995	3	ARPP21	1	35638945	35794496
139	ENSG00000173530	8	TNFRSF10D	-1	23135588	23164030
140	ENSG00000173898	11	SPTBN2	-1	66685248	66729226
141	ENSG00000174099	12	MSRB3	1	65278643	65491430
142	ENSG00000174407	20	MIR1-1HG	1	62550453	62570764
143	ENSG00000174672	11	BRSK2	1	1389899	1462689
144	ENSG00000175084	2	DES	1	219418377	219426739
145	ENSG00000175087	1	PDIK1L	1	26111165	26125543
146	ENSG00000175161	3	CADM2	1	84958981	86074429
147	ENSG00000176049	5	JAKMIP2	-1	147585439	147782848
148	ENSG00000177103	11	DSCAML1	-1	117427773	117817525
149	ENSG00000177508	16	IRX3	-1	54283304	54286763
150	ENSG00000177807	1	KCNJ10	-1	159998651	160070483
151	ENSG00000178445	9	GLDC	-1	6532464	6645783
152	ENSG00000179242	20	CDH4	1	61252426	61940617
153	ENSG00000179314	17	WSCD1	1	6057807	6124427
154	ENSG00000179915	2	NRXN1	-1	49918505	51225575
155	ENSG00000180287	1	PLD5	-1	242082986	242524696

156	ENSG00000182600	2	SNORC	1	232857270	232878708
157	ENSG00000183072	5	NKX2-5	-1	173232109	173235357
158	ENSG00000185155	1	MIXL1	1	226223618	226227054
159	ENSG00000185156	17	MFSD6L	-1	8797162	8799349
160	ENSG00000185565	3	LSAMP	-1	115802363	117139389
161	ENSG00000185627	11	PSMD13	1	236546	252984
162	ENSG00000185818	4	NAT8L	1	2059512	2069089
163	ENSG00000185973	X	TMLHE	-1	155490115	155669944
164	ENSG00000186231	6	KLHL32	1	96924620	97140754
165	ENSG00000187164	10	SHTN1	-1	116881482	117126586
166	ENSG00000187634	1	SAMD11	1	923928	944581
167	ENSG00000188015	1	S100A3	-1	153547329	153549372
168	ENSG00000188316	10	ENO4	1	116849512	116911788
169	ENSG00000188522	17	FAM83G	-1	18968789	19004804
170	ENSG00000196220	3	SRGAP3	-1	8980591	9363053
171	ENSG00000196338	X	NLGN3	1	71144831	71171201
172	ENSG00000196361	19	ELAVL3	-1	11451326	11481046
173	ENSG00000196376	6	SLC35F1	1	117907526	118317676
174	ENSG00000196581	1	AJAP1	1	4654732	4792534
175	ENSG00000196628	18	TCF4	-1	55222331	55664787
176	ENSG00000196767	X	POU3F4	1	83508261	83512127
177	ENSG00000197728	12	RPS26	1	56041351	56044697
178	ENSG00000198216	1	CACNA1E	1	181317690	181808084
179	ENSG00000198513	14	ATL1	1	50532509	50633068
180	ENSG00000198732	14	SMOC1	1	69854131	70032366
181	ENSG00000203930	X	LINC00632	1	140709562	140793215
182	ENSG00000204011	9	COL5A1-AS1	-1	134649385	134652843
183	ENSG00000204344	6	STK19	1	31971091	31982821
184	ENSG00000204624	1	DISP3	1	11479166	11537584

185	ENSG00000204683	10	C10orf113	-1	21125763	21146559
186	ENSG00000205758	21	CRYZL1	-1	33589341	33643926
187	ENSG00000213578	15	CPLX3	1	74826547	74831802
188	ENSG00000214160	3	ALG3	-1	184242301	184249548
189	ENSG00000214338	6	SOGA3	-1	127472794	127519191
190	ENSG00000214595	2	EML6	1	54723499	54972025
191	ENSG00000219438	22	FAM19A5	1	48489460	48850912
192	ENSG00000221818	8	EBF2	-1	25841730	26045397
193	ENSG00000235568	22	NFAM1	-1	42380410	42432395
194	ENSG00000237330	1	RNF223	-1	1070966	1074307
195	ENSG00000241370	6	RPP21	1	30345131	30346884
196	ENSG00000243232	5	PCDHAC2	1	140966235	141012344
197	ENSG00000243449	4	C4orf48	1	2041993	2043970
198	ENSG00000248383	5	PCDHAC1	1	140926369	141012344
199	ENSG00000253276	7	CCDC71L	-1	106656765	106660996
200	ENSG00000255537	11	AP000708.1	1	125495214	125499528

Table S3.4: Top 200 significant genes from the one-sided Wilcoxon test that compares brain-vs-brain scores and brain-vs-heart scores.

S3.6.7.1 Tissue-specific genes receive lower p-values in Wilcoxon test

From the boxplots in Figure S5, we can see that brain-specific genes and heart-specific genes receive lower p-values from the one-sided Wilcoxon test that compares brain-vs-brain alignment scores and brain-vs-heart alignment scores. When comparing heart-vs-heart scores and heart-vs-brain scores, heart-specific genes have much lower p-values. These results indicates that EpiAlign can correctly capture cell-type-characteristic chromatin state patterns.

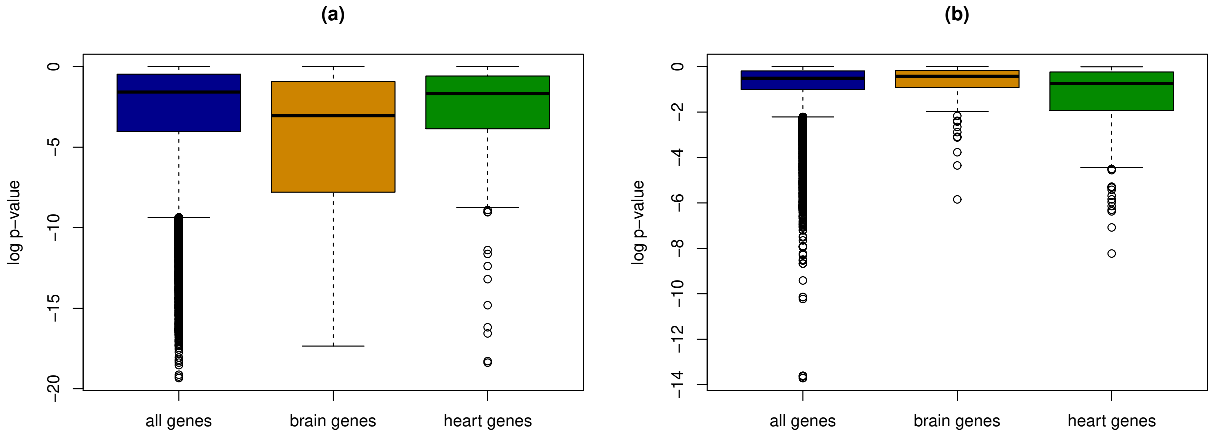


Figure 3.12: Boxplots of p-values of all genes, brain-specific genes and heart-specific genes from Wilcoxon tests which compares (a) brain-vs-brain scores with brain-vs-heart scores or (b) heart-vs-heart scores with brain-vs-heart scores.

S3.6.8 Motif analysis

The top 200 regions with the highest horizontal alignment scores and the cluster index to which each region belongs are listed in Table S3. We use the motif-discovery tool MEME and find that all the four clusters are characterized by certain motifs, the top motifs reported by MEME are:

Cluster 1: “ihihihihihihihihihio”; “edegbabg”; “aehihedhdeo”.

Cluster 2: “gegegogegegogogegege”; “gegegegegegegegegege”; gegegegegegegegegege”;

Cluster 3: “goglkjklnog”; “dededededededede”; “gegegegedegededegege”

Cluster 4: “oaogegegegbgege”; “mlklmlmlklklklk”; “nogogogo”.

	sample	chromosome	start.position	end.position	cluster
1	E003	chr22	22800001	23300000	1
2	E003	chr21	10700001	11200000	1
3	E003	chrX	61500001	62000000	1
4	E003	chr17	46400001	46900000	1
5	E003	chr20	59400001	59900000	1
6	E003	chr12	54100001	54600000	1

7	E003	chr12	37900001	38400000	1
8	E003	chr7	26900001	27400000	1
9	E003	chr2	176700001	177200000	1
10	E003	chr3	61500001	62000000	1
11	E003	chr6	157100001	157600000	1
12	E003	chr8	43400001	43900000	1
13	E003	chr16	33800001	34300000	1
14	E003	chr15	99100001	99600000	1
15	E003	chr11	2300001	2800000	1
16	E003	chrY	9900001	10400000	1
17	E003	chr2	92200001	92700000	1
18	E003	chr8	128600001	129100000	1
19	E003	chr7	61700001	62200000	2
20	E003	chr7	101300001	101800000	2
21	E003	chr5	89900001	90400000	2
22	E003	chr22	22300001	22800000	2
23	E003	chr4	78900001	79400000	2
24	E003	chr18	60100001	60600000	2
25	E003	chr10	34600001	35100000	2
26	E003	chr16	49400001	49900000	2
27	E003	chr6	15200001	15700000	2
28	E003	chr22	33900001	34400000	2
29	E003	chr5	54300001	54800000	2
30	E003	chr10	42200001	42700000	2
31	E003	chr14	89700001	90200000	2
32	E003	chr2	153200001	153700000	2
33	E003	chr2	121300001	121800000	2
34	E003	chr1	164300001	164800000	2
35	E003	chr19	37500001	38000000	2

36	E003	chr1	8400001	8900000	3
37	E003	chr4	190400001	190900000	2
38	E003	chr1	64200001	64700000	2
39	E003	chr5	106600001	107100000	2
40	E003	chr10	12000001	12500000	2
41	E003	chr5	13900001	14400000	2
42	E003	chr7	154900001	155400000	2
43	E003	chr11	31700001	32200000	2
44	E003	chr4	183900001	184400000	2
45	E003	chr12	132600001	133100000	2
46	E003	chr2	55200001	55700000	2
47	E003	chr8	131200001	131700000	2
48	E003	chr8	142200001	142700000	2
49	E003	chr17	59700001	60200000	2
50	E003	chr12	34400001	34900000	2
51	E003	chr11	12500001	13000000	2
52	E003	chr15	57300001	57800000	2
53	E003	chr20	49200001	49700000	2
54	E003	chr10	114400001	114900000	2
55	E003	chr5	87600001	88100000	2
56	E003	chr17	28800001	29300000	2
57	E003	chr22	29000001	29500000	2
58	E003	chr7	105300001	105800000	2
59	E003	chr22	17600001	18100000	2
60	E003	chr19	12800001	13300000	2
61	E003	chr12	32100001	32600000	2
62	E003	chr2	236200001	236700000	2
63	E003	chr3	185400001	185900000	2
64	E003	chr12	130400001	130900000	2

65	E003	chr15	26800001	27300000	2
66	E003	chr16	46000001	46500000	2
67	E003	chr11	107800001	108300000	2
68	E003	chr6	148900001	149400000	2
69	E003	chr4	93300001	93800000	2
70	E003	chr9	128200001	128700000	2
71	E003	chr15	50600001	51100000	2
72	E003	chr17	3900001	4400000	2
73	E003	chr1	235200001	235700000	2
74	E003	chr9	140300001	140800000	2
75	E003	chr19	1400001	1900000	2
76	E003	chr10	88400001	88900000	2
77	E003	chr15	28200001	28700000	2
78	E003	chr3	31600001	32100000	2
79	E003	chr2	188900001	189400000	2
80	E003	chr22	31800001	32300000	2
81	E003	chr15	44600001	45100000	2
82	E003	chr4	85400001	85900000	2
83	E003	chr13	98700001	99200000	2
84	E003	chr13	28300001	28800000	2
85	E003	chr9	33100001	33600000	2
86	E003	chr15	63800001	64300000	2
87	E003	chr1	39500001	40000000	2
88	E003	chr10	80500001	81000000	2
89	E003	chr3	121100001	121600000	2
90	E003	chr13	41000001	41500000	2
91	E003	chr16	89400001	89900000	2
92	E003	chr2	32400001	32900000	2
93	E003	chr1	219900001	220400000	2

94	E003	chr2	119400001	119900000	2
95	E003	chr1	10500001	11000000	2
96	E003	chr1	200200001	200700000	2
97	E003	chr12	112300001	112800000	2
98	E003	chr14	99600001	100100000	2
99	E003	chr8	30000001	30500000	2
100	E003	chr10	74800001	75300000	2
101	E003	chr13	113300001	113800000	2
102	E003	chr10	102800001	103300000	2
103	E003	chr18	52900001	53400000	2
104	E003	chr15	59300001	59800000	2
105	E003	chr16	1400001	1900000	2
106	E003	chr16	81200001	81700000	2
107	E003	chr8	102500001	103000000	2
108	E003	chr6	56300001	56800000	2
109	E003	chr13	100200001	100700000	2
110	E003	chr2	109000001	109500000	2
111	E003	chr10	126400001	126900000	2
112	E003	chr8	46800001	47300000	2
113	E003	chr11	126200001	126700000	2
114	E003	chr6	41100001	41600000	3
115	E003	chr2	102300001	102800000	3
116	E003	chr1	233000001	233500000	3
117	E003	chr9	16400001	16900000	3
118	E003	chr21	40200001	40700000	3
119	E003	chr9	130800001	131300000	3
120	E003	chr17	2600001	3100000	3
121	E003	chr10	96800001	97300000	3
122	E003	chrX	16700001	17200000	3

123	E003	chr13	100700001	101200000	3
124	E003	chr3	65500001	66000000	3
125	E003	chr17	25400001	25900000	3
126	E003	chr15	40100001	40600000	3
127	E003	chr5	64800001	65300000	3
128	E003	chr9	94800001	95300000	3
129	E003	chr9	124000001	124500000	3
130	E003	chr17	55300001	55800000	3
131	E003	chr22	43300001	43800000	3
132	E003	chr1	155500001	156000000	2
133	E003	chr9	125400001	125900000	2
134	E003	chr4	184500001	185000000	2
135	E003	chr20	35600001	36100000	3
136	E003	chr1	17700001	18200000	3
137	E003	chr9	70100001	70600000	3
138	E003	chr8	97400001	97900000	3
139	E003	chr22	40400001	40900000	3
140	E003	chr8	141600001	142100000	3
141	E003	chr12	11600001	12100000	3
142	E003	chr9	37600001	38100000	3
143	E003	chr8	102000001	102500000	3
144	E003	chr9	23700001	24200000	3
145	E003	chr22	45200001	45700000	3
146	E003	chr3	171700001	172200000	3
147	E003	chr15	90900001	91400000	3
148	E003	chr1	161800001	162300000	3
149	E003	chr15	42300001	42800000	3
150	E003	chr11	63600001	64100000	3
151	E003	chr1	21500001	22000000	3

152	E003	chr1	179800001	180300000	3
153	E003	chr5	46000001	46500000	3
154	E003	chr10	79200001	79700000	3
155	E003	chr18	55600001	56100000	3
156	E003	chr4	68300001	68800000	3
157	E003	chr7	121900001	122400000	3
158	E003	chr17	30200001	30700000	3
159	E003	chr11	61300001	61800000	3
160	E003	chr5	70500001	71000000	3
161	E003	chr2	202700001	203200000	3
162	E003	chr6	136500001	137000000	3
163	E003	chr1	23700001	24200000	3
164	E003	chr2	106100001	106600000	3
165	E003	chr4	48900001	49400000	3
166	E003	chr2	183700001	184200000	3
167	E003	chr6	21600001	22100000	2
168	E003	chr2	43400001	43900000	3
169	E003	chr16	72800001	73300000	3
170	E003	chr19	9200001	9700000	3
171	E003	chr1	32100001	32600000	3
172	E003	chr17	15600001	16100000	3
173	E003	chr17	27000001	27500000	3
174	E003	chr6	168200001	168700000	2
175	E003	chr6	37200001	37700000	2
176	E003	chr11	48000001	48500000	4
177	E003	chr10	7900001	8400000	4
178	E003	chr1	12300001	12800000	4
179	E003	chr9	111600001	112100000	4
180	E003	chr12	124900001	125400000	4

181	E003	chr7	55500001	56000000	4
182	E003	chr16	69000001	69500000	4
183	E003	chr15	43000001	43500000	4
184	E003	chr3	47400001	47900000	4
185	E003	chr10	70500001	71000000	4
186	E003	chr7	2400001	2900000	4
187	E003	chr4	7800001	8300000	4
188	E003	chr11	50300001	50800000	4
189	E003	chr6	166800001	167300000	4
190	E003	chr18	74600001	75100000	4
191	E003	chr5	139800001	140300000	4
192	E003	chr14	77300001	77800000	4
193	E003	chr11	121100001	121600000	4
194	E003	chr15	35000001	35500000	4
195	E003	chr6	29400001	29900000	4
196	E003	chr5	31400001	31900000	4
197	E003	chr7	23000001	23500000	4
198	E003	chr1	47800001	48300000	4
199	E003	chr10	13700001	14200000	4
200	E003	chr12	2900001	3400000	4

Table S3.5: The top 200 regions with the highest horizontal alignment scores are well partitioned into four clusters by average-linkage hierarchical clustering. The last column of the table is the cluster index to which each region belongs.

S3.6.9 GO analysis in cross-species application of EpiAlign

We obtain mouse-human homologous gene pairs from Ensembl BioMart and sort the mouse genes with lengths 200-400 kb by gene lengths and divide the homologous gene pairs into 12 groups each with 50 pairs. We look at the molecule function GO terms of the homolog pairs that have the highest alignment scores in each group. The result (Table S4) shows that homolougous genes with high alignment scores are also very similar in molecule function. The result also indicates that EpiAlign can identify homologous genes whose epigenetic patterns are more conserved in evolution, shedding new insights into translating scientific discoveries in mice into humans.

S3.6.10 Biological discovery based on top 8-mers

We have made the following discovery and have revised our manuscript on page 10 to discuss the potential improvement of EpiAlign: We have the following interesting findings when looking at the most common 8-mers in chromatin states identified by ChromHMM: we count the occurrence number of all the 8-mer strings from the epigenome of ESC sample E003. We look at the most frequently occurred 8-mer strings containing active TSS state (represented by ’a’). The most frequent 8-mer is ’aededede’, where ’e’ represents weak transcript and ’d’ represents strong transcript. This 8-mer can be interpreted as active gene region. However, there is also another 8-mer ’oioaoioi’, which frequently occurs but is hard to interpret. Here, ’o’ represents quiescent/low state and ’i’ represents heterochromatin. We select all ESC, heart and brain samples and inspected the overlap between known TSS and these two 8-mers. The results show that in all these samples, a high proportion (71% on average) of the 8-mer ’aededede’ discovered have an overlap with known TSS while only a small proportion (28% on average) of the 8-mer ’oioaoioi’ have an overlap with known TSS. This result indicates that some of the state ’a’ in ’oioaoioi’ may be noise.

Homologous pair		GO: molecule function	
Human	Mouse	Human	Mouse
AK5	AK5	nucleotide binding	nucleotide binding
GABRB2	GABRB2	transmembrane signaling receptor activity	transmembrane signaling receptor activity
CDH2	CDH2	calcium ion binding	calcium ion binding
GAB2	GAB2	transmembrane receptor protein tyrosine kinase adaptor activity	transmembrane receptor protein tyrosine kinase adaptor activity
NEB	NEB	actin binding	actin binding
CAMK4	CAMK4	nucleotide binding	nucleotide binding
PAPPA2	PAPPA2	metalloendopeptidase activity	metalloendopeptidase activity
CNTN1	CNTN1	protein binding	protein binding
DNAH7	DNAH7b	microtubule motor activity	microtubule motor activity
TMEM132C	TMEM132C	not available	not available
SPATA16	SPATA16	not available	not available
ADCY2	ADCY2	nucleotide binding	nucleotide binding

Homologous pair		GO: biological process	
Human	Mouse	Human	Mouse
AK5	AK5	nucleobase-containing compound metabolic process	nucleobase-containing compound metabolic process
GABRB2	GABRB2	ion transport	ion transport
CDH2	CDH2	cell morphogenesis	cell adhesion
GAB2	GAB2	transmembrane receptor protein tyrosine kinase signaling pathway	transmembrane receptor protein tyrosine kinase signaling pathway
NEB	NEB	muscle organ development	regulation of actin filament length
CAMK4	CAMK4	adaptive immune response	protein phosphorylation
PAPPA2	PAPPA2	regulation of cell growth	proteolysis
CNTN1	CNTN1	cell adhesion	cell adhesion
DNAH7	DNAH7b	microtubule-based movement	microtubule-based movement
TMEM132C	TMEM132C	not available	not available
SPATA16	SPATA16	not available	not available
ADCY2	ADCY2	renal water homeostasis c	AMP biosynthetic process

Table S3.6: The top 1 GO terms (both molecule function and biological process) of the homolog pairs that have the highest alignment scores in each group.

CHAPTER 4

Summary and future directions

In this dissertation, I introduced two statistical methods Clipper and EpiAlign, which I developed for high-throughput data analyses that compare two conditions. Below I summarize the two methods and list the future research directions for each method.

4.1 P-value-free FDR control on high-throughput data from two conditions

In Chapter 2, we proposed the Clipper, a p-value-free FDR control framework, for identifying interesting features by contrasting high-throughput data under two conditions. Clipper makes FDR control more reliable and flexible by avoiding the use of p-values, which are based on assumptions likely to be violated in real data analysis. We verified the FDR control by Clipper in comprehensive simulation studies and two real data analyses: peak calling from ChIP-seq data and DEG identification from RNA-seq data. Our results indicate that Clipper can improve the reliability of FDR control and thus the reproducibility of scientific discoveries.

In most bioinformatics method papers, the FDR control was merely assumed by relying on p-values but rarely validated. However, p-values were often invalid when model assumptions were violated or the p-value calculation was problematic. By proposing Clipper, we would also like to voice the importance of validating the FDR control in bioinformatics method development.

The first future direction of Clipper is to explore the power of different contrast scores. As the core component of Clipper, contrast score is calculated for each feature to summarize the difference between the feature's measurements under the two conditions. Clipper currently uses two contrast scores, minus and maximum, and an advantage of Clipper is that it allows other definitions of contrast scores. We would like to explore alternative contrast scores and their power with respect to data characteristics and analysis tasks. For example, Clipper currently only focuses on the difference in means. With other definitions of contrast scores, we may distinguish two conditions when they have the same mean but different distributions. We may further design contrast scores based on multivariate test statistics so that Clipper can identify interesting features from more than one perspective (e.g., differences in mean and variance) at the same time. Furthermore, we may generalize Clipper to be robust against sample batch effects by constructing the contrast score as a regression-based test statistic that has batch effects removed.

Second, Clipper currently only focus on the frequentist FDR. A possible generalization of Clipper is to consider the Bayesian framework so that we can leverage prior knowledge of features to increase the power for identifing interesting features.

Third, we would also like to escalate Clipper into standalone bioinformatics methods for specific data analyses, for which data processing and characteristics (e.g., peak lengths, GC contents, zero proportions, and batch effects) must be appropriately accounted for before Clipper is used for the FDR control.

Finally, we would like to explore possible generalization of Clipper to features identification across more than two conditions. To tailor Clipper for such analysis, we could define a new contrast score that differentiates the genes with stationary expression (uninteresting features) from the other genes with varying expression (interesting features). Further studies are needed to explore the possibility of extending Clipper to the regression framework so that Clipper can accommodate data of multiple conditions or even continuous conditions, as well as adjusting for confounding covariates.

4.2 Alignment-based bioinformatic tool for comparing chromatin state sequences

In Chapter 3, we proposed the EpiAlign algorithm for aligning chromatin state sequences inferred from multi-track epigenomic signals. EpiAlign can be a powerful tool for studying the epigenetic dynamics across multiple epigenomes, or between two conditions.

For future directions, first, our current alignment results are based on ChromHMM, which summarizes multi-track epigenomic signals into one track. There are other tools for chromatin pattern discovery, such as Segway [15], EpiCSeg [85], and IDEAS [87]. It would be interesting to compare these tools with ChromHMM and analyze how their inferred chromatin states would affect the alignment results of EpiAlign.

Second, we can improve the compression of chromatin state sequences, an important step in EpiAlign. We added the compression step to capture and extract the dynamics of chromatin states among biological samples. However, previous studies have implied that the length information of certain chromatin states is important for comparing a region's chromatin states under different conditions [105–107]. Future refinement of the compression step, or refinement of length information usage after compression, should be considered.

Finally, in some computationally efficient sequence alignment algorithms, hash tables or tree-based data structures are utilized to index the database, and these techniques have greatly increased the efficiency of query retrieval. EpiAlign can benefit from similar techniques to improve its computation efficiency.

4.3 Combination of Clipper and EpiAlign for identifying conserved epigenomic signals between two conditions

Another future direction is the combination of Clipper and EpiAlign. In the vertical alignment analysis of EpiAlign, we compared two sets of alignment scores: pairwise alignment scores within the same condition (gender or cell type), and pairwise alignment scores between two conditions. We compared these two sets of alignments scores for each protein-coding gene. To identify the genes whose two sets of alignment scores are most differential, we performed the rank-sum Wilcoxon test, which does not have any distributional assumption on the alignment scores in each set but does not have good power when the alignment scores are few (i.e., not many epigenome samples are available in each condition). In this case, Clipper can serve as a good alternative to the Wilcoxon test for identifying the genes with differential alignment scores under an FDR threshold.

Bibliography

[1] Eric D Green et al. “Strategic vision for improving human health at The Forefront of Genomics”. In: *Nature* 586.7831 (2020), pp. 683–692.

[2] Stanislaw Supplitt et al. “Current Achievements and Applications of Transcriptomics in Personalized Cancer Medicine”. In: *International Journal of Molecular Sciences* 22.3 (2021), p. 1422.

[3] Ali Khodadadian et al. “Genomics and transcriptomics: the powerful technologies in precision medicine”. In: *International Journal of General Medicine* 13 (2020), p. 627.

[4] Peter J Park. “ChIP–seq: advantages and challenges of a maturing technology”. In: *Nature Reviews Genetics* 10.10 (2009), pp. 669–680.

[5] Pamela J Mitchell and Robert Tjian. “Transcriptional regulation in mammalian cells by sequence-specific DNA binding proteins”. In: *Science* 245.4916 (1989), pp. 371–378.

[6] Mark Ptashne and Alexander Gann. “Transcriptional activation by recruitment”. In: *Nature* 386.6625 (1997), pp. 569–577.

[7] Yoav Benjamini and Yosef Hochberg. “Controlling the false discovery rate: a practical and powerful approach to multiple testing”. In: *Journal of the Royal statistical society: series B (Methodological)* 57.1 (1995), pp. 289–300.

[8] John D Storey. “A direct approach to false discovery rates”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 64.3 (2002), pp. 479–498.

[9] Yoav Benjamini and Yosef Hochberg. “Multiple hypotheses testing with weights”. In: *Scandinavian Journal of Statistics* 24.3 (1997), pp. 407–418.

[10] Nikolaos Ignatiadis et al. “Data-driven hypothesis weighting increases detection power in genome-scale multiple testing”. In: *Nature methods* 13.7 (2016), pp. 577–580.

[11] Lihua Lei and William Fithian. “Adapt: an interactive procedure for multiple testing with side information”. In: *arXiv preprint arXiv:1609.06035* (2016).

[12] Simina M Boca and Jeffrey T Leek. "A direct approach to estimating false discovery rates conditional on covariates". In: *PeerJ* 6 (2018), e6035.

[13] Xinzhou Ge et al. "Clipper: p-value-free FDR control on high-throughput data from two conditions". In: *bioRxiv* (2020).

[14] Jason Ernst and Manolis Kellis. "ChromHMM: automating chromatin-state discovery and characterization". In: *Nature methods* 9.3 (2012), p. 215.

[15] Michael M Hoffman et al. "Unsupervised pattern discovery in human chromatin structure through genomic segmentation". In: *Nature methods* 9.5 (2012), p. 473.

[16] Tarjei S Mikkelsen et al. "Genome-wide maps of chromatin state in pluripotent and lineage-committed cells". In: *Nature* 448.7153 (2007), p. 553.

[17] Maria E Figueroa et al. "DNA methylation signatures identify biologically distinct subtypes in acute myeloid leukemia". In: *Cancer cell* 17.1 (2010), pp. 13–27.

[18] Jason Ernst et al. "Mapping and analysis of chromatin state dynamics in nine human cell types". In: *Nature* 473.7345 (2011), p. 43.

[19] Nathaniel D Heintzman et al. "Histone modifications at human enhancers reflect global cell-type-specific gene expression". In: *Nature* 459.7243 (2009), p. 108.

[20] Anshul Kundaje et al. "Integrative analysis of 111 reference human epigenomes". In: *Nature* 518.7539 (2015), p. 317.

[21] Wei Vivian Li, Zahra S Razaee, and Jingyi Jessica Li. "Epigenome overlap measure (EPOM) for comparing tissue/cell types based on chromatin states". In: *BMC genomics*. Vol. 17. 1. BioMed Central. 2016, S10.

[22] Saul B Needleman and Christian D Wunsch. "A general method applicable to the search for similarities in the amino acid sequence of two proteins". In: *Journal of molecular biology* 48.3 (1970), pp. 443–453.

[23] TF SMITH and MS Waterman. "Identification of Common Molecular Subsequence". In: *Journal of Molecular Biology* 147 (1981), pp. 195–197.

[24] Stephen F Altschul et al. "Basic local alignment search tool". In: *Journal of molecular biology* 215.3 (1990), pp. 403–410.

[25] Yiling Chen. "Statistical criteria and procedures for controlling false positives with applications to biological and biomedical data analysis". PhD thesis. University of California, Los Angeles, 2021.

[26] Yong Zhang et al. "Model-based analysis of ChIP-Seq (MACS)". In: *Genome biology* 9.9 (2008), pp. 1–9.

[27] Sven Heinz et al. "Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities". In: *Molecular cell* 38.4 (2010), pp. 576–589.

[28] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. "edgeR: a Bioconductor package for differential expression analysis of digital gene expression data". In: *Bioinformatics* 26.1 (2010), pp. 139–140.

[29] Michael I Love, Wolfgang Huber, and Simon Anders. "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2". In: *Genome biology* 15.12 (2014), p. 550.

[30] Cole Trapnell et al. "Differential analysis of gene regulation at transcript resolution with RNA-seq". In: *Nature biotechnology* 31.1 (2013), pp. 46–53.

[31] Jun Li et al. "Normalization, testing, and false discovery rate estimation for RNA-sequencing data". In: *Biostatistics* 13.3 (2012), pp. 523–538.

[32] Thomas J Hardcastle and Krystyna A Kelly. "baySeq: empirical Bayesian methods for identifying differential expression in sequence count data". In: *BMC bioinformatics* 11.1 (2010), pp. 1–14.

[33] Gordon K Smyth. "Linear models and empirical bayes methods for assessing differential expression in microarray experiments". In: *Statistical applications in genetics and molecular biology* 3.1 (2004), pp. 1–25.

[34] Matthew E Ritchie et al. “limma powers differential expression analyses for RNA-sequencing and microarray studies”. In: *Nucleic acids research* 43.7 (2015), e47–e47.

[35] Bradley Efron and Robert Tibshirani. “Empirical Bayes methods and false discovery rates for microarrays”. In: *Genetic epidemiology* 23.1 (2002), pp. 70–86.

[36] Bradley Efron et al. “Empirical Bayes analysis of a microarray experiment”. In: *Journal of the American statistical association* 96.456 (2001), pp. 1151–1160.

[37] Matthew Stephens. “False discovery rates: a new deal”. In: *Biostatistics* 18.2 (2017), pp. 275–294.

[38] John D Storey and Robert Tibshirani. “Statistical significance for genomewide studies”. In: *Proceedings of the National Academy of Sciences* 100.16 (2003), pp. 9440–9445.

[39] Anat Reiner, Daniel Yekutieli, and Yoav Benjamini. “Identifying differentially expressed genes using false discovery rate controlling procedures”. In: *Bioinformatics* 19.3 (2003), pp. 368–375.

[40] Bing Yang et al. “Identification of cross-linked peptides from complex samples”. In: *Nature methods* 9.9 (2012), pp. 904–906.

[41] James Robert White, Niranjan Nagarajan, and Mihai Pop. “Statistical methods for detecting differentially abundant features in clinical metagenomic samples”. In: *PLoS Comput Biol* 5.4 (2009), e1000352.

[42] Andrey A Shabalin. “Matrix eQTL: ultra fast eQTL analysis via large matrix operations”. In: *Bioinformatics* 28.10 (2012), pp. 1353–1358.

[43] Stijn Hawinkel et al. “A broken promise: microbiome differential abundance methods do not control the false discovery rate”. In: *Briefings in bioinformatics* 20.1 (2019), pp. 210–221.

[44] Ye Zheng and Sündüz Keleş. “FreeHi-C simulates high-fidelity Hi-C data for benchmarking and data augmentation”. In: *Nature Methods* 17.1 (2020), pp. 37–40.

[45] Joses Ho et al. “Moving beyond P values: data analysis with estimation graphics”. In: *Nature methods* 16.7 (2019), pp. 565–566.

[46] Dongyuan Song and Jingyi Jessica Li. “PseudotimeDE: inference of differential gene expression along cell pseudotime with well-calibrated p-values from single-cell RNA sequencing data”. In: *bioRxiv* (2020).

[47] Virginia Goss Tusher, Robert Tibshirani, and Gilbert Chu. “Significance analysis of microarrays applied to the ionizing radiation response”. In: *Proceedings of the National Academy of Sciences* 98.9 (2001), pp. 5116–5121.

[48] Jesse Hemerik and Jelle J Goeman. “False discovery proportion estimation by permutations: confidence for significance analysis of microarrays”. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 80.1 (2018), pp. 137–155.

[49] Jesse Hemerik, Aldo Solari, and Jelle J Goeman. “Permutation-based simultaneous confidence bounds for the false discovery proportion”. In: *Biometrika* 106.3 (2019), pp. 635–649.

[50] Rina Foygel Barber and Emmanuel J Candès. “Controlling the false discovery rate via knockoffs”. In: *The Annals of Statistics* 43.5 (2015), pp. 2055–2085.

[51] Ery Arias-Castro and Shiyun Chen. “Distribution-free multiple testing”. In: *Electronic Journal of Statistics* 11.1 (2017), pp. 1983–2001.

[52] Yoav Benjamini. “Selective inference: The silent killer of replicability”. In: *Issue 2.4* 2.4 (2020).

[53] Kristen Emery et al. “Multiple Competition-Based FDR Control and Its Application to Peptide Detection”. In: *International Conference on Research in Computational Molecular Biology*. Springer. 2020, pp. 54–71.

[54] Jaime Roquero Gimenez and James Zou. “Improving the Stability of the Knockoff Procedure: Multiple Simultaneous Knockoffs and Entropy Maximization”. In: *arXiv preprint arXiv:1810.11378* (2018).

[55] Daniel Yekutieli and Yoav Benjamini. "Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics". In: *Journal of Statistical Planning and Inference* 82.1-2 (1999), pp. 171–196.

[56] Timothy Bailey et al. "Practical guidelines for the comprehensive analysis of ChIP-seq data". In: *PLoS Comput Biol* 9.11 (2013), e1003326.

[57] ENCODE Project Consortium et al. "An integrated encyclopedia of DNA elements in the human genome". In: *Nature* 489.7414 (2012), pp. 57–74.

[58] Vishaka Datta, Sridhar Hannenhalli, and Rahul Siddharthan. "ChIPulate: A comprehensive ChIP-seq simulation pipeline". In: *PLoS computational biology* 15.3 (2019), e1006921.

[59] Aaron Diaz et al. "Normalization, bias correction, and peak calling for ChIP-seq". In: *Statistical applications in genetics and molecular biology* 11.3 (2012).

[60] Mark D Robinson and Alicia Oshlack. "A scaling normalization method for differential expression analysis of RNA-seq data". In: *Genome biology* 11.3 (2010), pp. 1–9.

[61] Claire R Williams et al. "Empirical assessment of analysis workflows for differential expression analysis of human samples using RNA-Seq". In: *BMC bioinformatics* 18.1 (2017), p. 38.

[62] Marek Gierliński et al. "Statistical models for RNA-seq data derived from a two-condition 48-replicate experiment". In: *Bioinformatics* 31.22 (2015), pp. 3625–3630.

[63] Keegan Korthauer et al. "A practical guide to methods controlling false discoveries in computational biology". In: *Genome biology* 20.1 (2019), pp. 1–21.

[64] Qunhua Li et al. "Measuring reproducibility of high-throughput experiments". In: *The annals of applied statistics* 5.3 (2011), pp. 1752–1779.

[65] Guangchuang Yu et al. "clusterProfiler: an R package for comparing biological themes among gene clusters". In: *Omics: a journal of integrative biology* 16.5 (2012), pp. 284–287.

[66] Greg Finak et al. “MAST: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell RNA sequencing data”. In: *Genome biology* 16.1 (2015), pp. 1–13.

[67] Xiaojie Qiu et al. “Single-cell mRNA quantification and differential analysis with Census”. In: *Nature methods* 14.3 (2017), pp. 309–315.

[68] Charlotte Soneson and Mark D Robinson. “Bias, robustness and scalability in single-cell differential expression analysis”. In: *Nature methods* 15.4 (2018), p. 255.

[69] Tianyi Sun et al. “scDesign2: an interpretable simulator that generates high-fidelity single-cell gene expression count data with gene correlations captured”. In: *bioRxiv* (2020).

[70] Jiarui Ding et al. “Systematic comparison of single-cell and single-nucleus RNA-sequencing methods”. In: *Nature biotechnology* 38.6 (2020), pp. 737–746.

[71] Grace XY Zheng et al. “Massively parallel digital transcriptional profiling of single cells”. In: *Nature communications* 8.1 (2017), pp. 1–12.

[72] Evan Z Macosko et al. “Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets”. In: *Cell* 161.5 (2015), pp. 1202–1214.

[73] Ning Wang et al. “Identifying the combinatorial control of signal-dependent transcription factors”. In: *PLOS Computational Biology* 17.6 (2021), e1009095.

[74] Jonathan Thorsen et al. “Large-scale benchmarking reveals false discoveries and count transformation sensitivity in 16S rRNA gene amplicon data analysis methods used in microbiome studies”. In: *Microbiome* 4.1 (2016), p. 62.

[75] Kun He et al. “Null-free False Discovery Rate Control Using Decoy Permutations for Multiple Testing”. In: *arXiv preprint arXiv:1804.08222* (2018).

[76] Richard A Young. “Control of the embryonic stem cell state”. In: *Cell* 144.6 (2011), pp. 940–954.

[77] Chikara Furusawa and Kunihiko Kaneko. “A dynamical-systems view of stem cell biology”. In: *Science* 338.6104 (2012), pp. 215–217.

[78] Julia Ye and Robert Blelloch. "Regulation of pluripotency by RNA binding proteins". In: *Cell stem cell* 15.3 (2014), pp. 271–280.

[79] Matteo Pellegrini and Roberto Ferrari. "Epigenetic analysis: ChIP-chip and ChIP-seq". In: *Next Generation Microarray Bioinformatics.* Springer, 2012, pp. 377–387.

[80] Bradley E Bernstein et al. "The NIH roadmap epigenomics mapping consortium". In: *Nature biotechnology* 28.10 (2010), p. 1045.

[81] Hendrik G Stunnenberg et al. "The International Human Epigenome Consortium: a blueprint for scientific collaboration and discovery". In: *Cell* 167.5 (2016), pp. 1145–1149.

[82] Pengfei Yu et al. "Spatiotemporal clustering of the epigenome reveals rules of dynamic gene regulation". In: *Genome research* (2012).

[83] Jacob Biesinger, Yuanfeng Wang, and Xiaohui Xie. "Discovering and mapping chromatin states using a tree hidden Markov model". In: *BMC bioinformatics.* Vol. 14. 5. BioMed Central. 2013, S4.

[84] Benedikt Zacher et al. "Annotation of genomics data using bidirectional hidden Markov models unveils variations in Pol II transcription cycle". In: *Molecular systems biology* 10.12 (2014), p. 768.

[85] Alessandro Mammana and Ho-Ryun Chung. "Chromatin segmentation based on a probabilistic model for read counts explains a large portion of the epigenome". In: *Genome biology* 16.1 (2015), p. 151.

[86] Jimin Song and Kevin C Chen. "Spectacle: fast chromatin state annotation using spectral learning". In: *Genome biology* 16.1 (2015), p. 33.

[87] Yu Zhang et al. "Jointly characterizing epigenetic dynamics across multiple human cell types". In: *Nucleic acids research* 44.14 (2016), pp. 6721–6731.

[88] Benedikt Zacher et al. "Accurate promoter and enhancer identification in 127 ENCODE and roadmap epigenomics cell types and tissues by GenoSTAN". In: *PloS one* 12.1 (2017), e0169249.

[89] Xiong Liu et al. "TiGER: a database for tissue-specific gene expression and regulation". In: *BMC bioinformatics* 9.1 (2008), p. 271.

[90] Ryan Lister et al. "Human DNA methylomes at base resolution show widespread epigenomic differences". In: *nature* 462.7271 (2009), p. 315.

[91] Jennifer Harrow et al. "GENCODE: the reference human genome annotation for The ENCODE Project". In: *Genome research* 22.9 (2012), pp. 1760–1774.

[92] Susanne Arnold, Gilda Wright de Araujo, and Cordian Beyer. "Gender-specific regulation of mitochondrial fusion and fission gene transcription and viability of cortical astrocytes by steroid hormones". In: *Journal of molecular endocrinology* 41.5 (2008), pp. 289–300.

[93] Eran Eden et al. "GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists". In: *BMC bioinformatics* 10.1 (2009), p. 48.

[94] Yang Yang et al. "Large-scale mapping of mammalian transcriptomes identifies conserved genes associated with different cell states". In: *Nucleic acids research* 45.4 (2017), pp. 1657–1672.

[95] W James Kent et al. "The human genome browser at UCSC". In: *Genome research* 12.6 (2002), pp. 996–1006.

[96] Timothy L Bailey, Charles Elkan, et al. "Fitting a mixture model by expectation maximization to discover motifs in bipolymers". In: (1994).

[97] Jason Ernst and Manolis Kellis. "Large-scale imputation of epigenomic datasets for systematic annotation of diverse human tissues". In: *Nature biotechnology* 33.4 (2015), p. 364.

[98] Maartje J Vogel et al. "Human heterochromatin proteins form large domains containing KRAB-ZNF genes". In: *Genome research* 16.12 (2006), pp. 000–000.

[99] Philippa Melamed et al. "Transcriptional enhancers: Transcription, function and flexibility". In: *Transcription* 7.1 (2016), pp. 26–31.

[100] Phillippa C Taberlay et al. "Polycomb-repressed genes have permissive enhancers that initiate reprogramming". In: *Cell* 147.6 (2011), pp. 1283–1294.

[101] Feng Yue et al. "A comparative encyclopedia of DNA elements in the mouse genome". In: *Nature* 515.7527 (2014), p. 355.

[102] Roman L Tatusov, Eugene V Koonin, and David J Lipman. "A genomic perspective on protein families". In: *Science* 278.5338 (1997), pp. 631–637.

[103] Daniel R Zerbino et al. "Ensembl 2018". In: *Nucleic acids research* 46.D1 (2017), pp. D754–D761.

[104] Elizabeth M Blackwood and James T Kadonaga. "Going the distance: a current view of enhancer action". In: *Science* 281.5373 (1998), pp. 60–63.

[105] Kaifu Chen et al. "Broad H3K4me3 is associated with increased transcription elongation and enhancer activity at tumor-suppressor genes". In: *Nature genetics* 47.10 (2015), p. 1149.

[106] John Arne Dahl et al. "Broad histone H3K4me3 domains in mouse oocytes modulate maternal-to-zygotic transition". In: *Nature* 537.7621 (2016), p. 548.

[107] Xiaoyu Liu et al. "Distinct features of H3K4me3 and H3K27me3 chromatin domains in pre-implantation embryos". In: *Nature* 537.7621 (2016), p. 558.

[108] Yu He and Ting Wang. "EpiCompare: an online tool to define and explore genomic regions with tissue or cell type-specific epigenomic features". In: *Bioinformatics* 33.20 (2017), pp. 3268–3275.

[109] Angela Yen and Manolis Kellis. "Systematic chromatin state comparison of epigenomes associated with diverse properties including sex and tissue type". In: *Nature communications* 6 (2015), p. 7973.