Genome Biology

SOFTWARE

Open Access

Enhlink infers distal and context-specific enhancer–promoter linkages

Olivier B. Poirion[1*], Wulin Zuo[1], Catrina Spruce[2], Candice N. Baker[2], Sandra L. Daigle[2], Ashley Olson[2,3], Daniel A. Skelly[2], Elissa J. Chesler[2,3], Christopher L. Baker[2,3†] and Brian S. White[1†]

†Christopher L. Baker and Brian S. White contributed equally to this work.

*Correspondence: o.poirion@gmail.com

[1] The Jackson Laboratory for Genomic Medicine, Farmington, CT, USA
[2] The Jackson Laboratory, Bar Harbor, ME, USA
[3] Center for Systems Neurogenetics of Addiction at The Jackson Laboratory, Bar Harbor, ME, USA

Abstract

Enhlink is a computational tool for scATAC-seq data analysis, facilitating precise interrogation of enhancer function at the single-cell level. It employs an ensemble approach incorporating technical and biological covariates to infer condition-specific regulatory DNA linkages. Enhlink can integrate multi-omic data for enhanced specificity, when available. Evaluation with simulated and real data, including multi-omic datasets from the mouse striatum and novel promoter capture Hi-C data, demonstrate that Enhlink outperfoms alternative methods. Coupled with eQTL analysis, it identified a putative super-enhancer in striatal neurons. Overall, Enhlink offers accuracy, power, and potential for revealing novel biological insights in gene regulation.

Keywords: Single-cell, Linkage analysis, Enhancers inference, Chromatin accessibility, Machine-learning

Introduction

Gene transcription is regulated by non-coding DNA elements called enhancers. Each consists of dense clusters of recognition motifs for sequence- and cell type-specific transcription factors (TFs), which bind and subsequently recruit coregulators, chromatin remodelers and modifiers, and RNA polymerase II [1]. A single gene can be regulated by multiple enhancers, with the cell type-specific activity of its enhancers conferring spatiotemporal control over it [2, 3]. Enhancer disruption and its concomitant modulation of target gene expression are increasingly recognized as disease-causing mechanisms [4, 5]. In complex diseases, > 90% of single nucleotide polymorphisms (SNPs) identified by genome-wide association studies (GWAS) are in non-coding regions of the genome far from promoters and potentially within enhancers [5]. The link between an enhancer and its target gene (or, equivalently, promoter) needs to be established and is an ongoing challenge in the field.

Enhancer–promoter links can be directly detected with experimental techniques including Hi-C [6, 7]. However, complex protocols, high cost, low resolution [8],

and their inability to detect interchromosomal interactions [1] currently limit their applications.

As an alternative, Pliner and colleagues demonstrated how links can be inferred *computationally* by exploiting measurements of chromatin accessibility from single-cell ATAC-seq (scATAC-seq) data at a gene's promoter and its active enhancers during transcription. The authors first inferred open chromatin regions (OCRs) from "peaks" of reads in scATAC-seq data and applied a computational method, Cicero, to identify enhancer–promoter pairs with correlated peaks of chromatin accessibility [9]. Cicero handles the sparsity of scATAC-seq data by aggregating binary accessibility data from similar cells into counts, with related cells determined through similarities in a low-dimensional embedding. It reduces batch effects, principally arising from library size, by adjusting aggregated counts. Finally, it addresses the high dimension inherent in genome-wide discovery by inferring regularized covariance matrices describing accessibility peaks.

Following Cicero's pioneering approach, linkage inference from scATAC-seq has become a popular strategy in various exploratory [10, 1112] and methodological studies [13]. Several other recent methods can also be used to infer enhancer–gene links from scATAC-seq data. Signac [14], ArchR [15], and SnapATAC [16] are comprehensive toolkits for scATAC-seq analysis that include linkage inference methods. In contrast, Robustlink [17] is a novel approach specifically tailored for linkage inference. ArchR computes the Pearson correlation between accessible regions represented in a low-dimensional embedding of aggregated cells and derives a p-value from it. Signac also computes Pearson correlation p-values, but instead does so using a random background of enhancers. SnapATAC fits univariate logistic regression models to chromatin accessibility using gene expression as features. Finally, Robustlink creates *meta-cells* by aggregating cells from graph communities and infers permuted score distributions to derive p-values from the correlation scores. Robustlink, SnapATAC, and Signac are designed to associate enhancer accessibilities with gene expression from either a multi-omic dataset or a matching scRNA-seq dataset which is combined with the scATAC-seq using a label transfer procedure [16]. In contrast, ArchR has the ability to process scATAC-seq alone or in combination with scRNA-seq. To summarize, these approaches leverage correlations between enhancers and promoter accessibility or gene expression at the single-cell level to deduce enhancer–gene links.

However, single-cell experiments have continued to grow in size and complexity since these methods were developed, not only leading to exquisite contextual specificity for inference on enhancer–promoter interactions but also producing additional difficulties that are not adequately addressed by existing computational methods. For example, our recent murine type 2 diabetes (T2D) study made use of a factorial and hierarchical experimental design to characterize the contribution of genetics, sex, and diet to cellular heterogeneity in two metabolism-related tissues (Poirion et al., 2024 [18]), by performing large-scale, single-cell sequencing across technical batches. Existing methods can not directly model the impact of biological covariates nor can cell-binning approaches control for technical covariates that differ across cells *within* a bin.

To address the challenges of studies with complex experimental designs, we developed Enhlink, a novel approach for inferring enhancer–promoter co-accessibility. It detects

biological effects and controls technical effects by incorporating appropriate covariates into a nonlinear modeling framework involving single cells, rather than aggregates. It selects a parsimonious set of enhancers associated with a promoter to smooth the sparse representation of any individual enhancer while prioritizing those with the largest effect. To do so, Enhlink uses a random forest-like approach, where cell-level (binary) accessibilities of enhancers and biological and technical factors are features and the cell-level accessibility of a promoter is the response variable. If multi-omic chromatin accessibility and gene expression measurements are simultaneously available for each cell, Enhlink can further prioritize enhancers by associating them with the expression of the promoter's target gene. Unlike existing methods, Enhlink has the ability to predict both proximal and distal enhancer–gene linkages and identify linkage specific to biological covariates, while also integrating a simulation workflow that utilizes experimentally validated enhancer–promoter signals to optionally estimate prediction accuracy.

Using simulation parameterized by experimentally validated enhancer–promoter pairs, we show that Enhlink minimizes false positives and negatives relative to other approaches. We further demonstrate that Enhlink results are resilient to technical batches in our T2D study and that it has superior precision evaluated using enhancer–promoter interactions detected from paired promoter capture (PC)Hi-C data. Finally, we generated a multi-ome single-nuclei (sn)ATAC- and RNA-seq dataset from a study of sex and strain differences in the mouse striatum, a brain region involved in motivated learning and associated with acute and chronic effects of drug addiction [19, 20]. After identifying the two main neuron populations defined by their expression of dopamine receptor 1 (*Drd1*) or 2 (*Drd2*), we inferred neuron-specific enhancer–promoter links using both promoter accessibility and gene expression. We identified strong putative *cis* and *trans*regulatory regions among the two classes of neurons that we also intersected with a set of genetic variants. Notably, we identified several enhancers 500 kb downstream of the *Drd1* promoter that were directly correlated with the regulation of multiple distal genes involved in the Drd1/Drd2 genetic program and may act as a super-enhancer. Enhlink should similarly enable the discovery of enhancer–promoter co-accessibility in other complex scATAC-seq and multi-omic snATAC-/snRNA-seq datasets.

Results

Inferring biologically meaningful co-accessibilities from sc/snATAC-seq data

To assess enhancer–promoter co-accessibility inference from snATAC-seq data, we used the results of a previously published human heart study (CARE) that generated snATAC-seq data and experimentally validated enhancer–promoter pairs [21]. We focused on the *KCNH2* promoter, for which the study identified an enhancer with a nearby risk variant (rs7789146) (Fig. 1A) linked to atrial fibrillation. That study validated the role of this variant on enhancer function via CRISPR-Cas9 genome editing of a human pluripotent stem cell-derived cardiomyocyte (CM) cell line [21]. Here, we determined whether the accessibilities of the rs7789146 enhancer and *KCNH2* promoter correlated across cells, by representing each as a Boolean vector whose entries reflect whether at least one snATAC-seq read was present within the respective region and cell. We then computed the accuracy, recall, and f1 score between the promoter and the enhancer vectors,

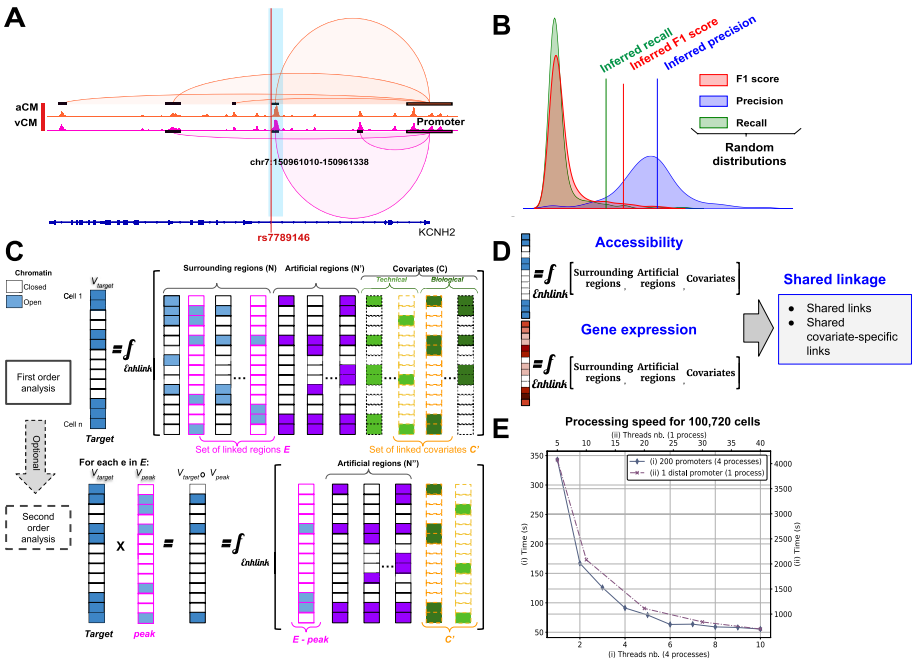

Fig. 1 Enhlink infers linkage by modeling covariates, clusters and the surrounding enhancers. **A** Chromatin accessibility tracks with enhancer–promoter co-accessibility links inferred with Enhlink from human atrial (aCM) and ventricular (vCM) cardiomyocytes. The enhancer highlighted in blue was previously experimentally validated. **B** Accuracy (f1-score, precision, recall) scores computed from validated vCM enhancer/promoter pair for the promoter of *KCNH2* using scATAC-seq data and compared to distributions of f1-scores, precisions, recalls obtained from random enhancers. High f1-score indicates that overall cells have similar accessibilities at the promoter and the enhancer. **C** Enhlink models a target region as a function of its surrounding genomic regions (i.e., enhancers) and biological and technical covariates. Artificial regions are added to reach a sufficient number of variables for computing feature scores and *p*-values (*t*-tests). Enhlink can optionally perform a second-order analysis to identify covariates associated with links. **D** Enhlink can leverage multi-omics datasets by modelling a target region by either its accessibility or its expression and by intersecting the two resulting sets to identify links shared across both modalities. **E** Processing time for detecting associations (scenario I) for 200 promoters and their cis (+/− 250 kb) OCR features from the islet dataset using four processes and (scenario II) between 1 promoter and 260,344 *cis* and *trans* OCR features using one process. Processing time (left axis for I and right for II) as a function of number of threads per process (bottom axis for I and top for II)

separately for atrial (aCM) and ventricular (vCM) cardiomyocytes and compared them to null score distributions (Fig. 1B and S1; see "Methods"). We expanded this analysis to the promoter of *MYL2* and three of its putative enhancers, also highlighted by the original study [21] (Figure S1). In all cases, the f1-score and recall values were significantly higher than random ($p < 0.05$; Fig. 1B and S1). These results further justify computational methods that infer enhancer–promoter links from their co-accessibility in snATAC-seq data, while the enhancers associated with the *KCNH2* and *MYL2* promoters provide a means of evaluating such methods.

Multi-omic inference of condition-specific enhancer–promoter links with Enhlink

Enhlink is a new approach that has been designed as an efficient computational framework for inferring co-accessibilities between OCRs, such as enhancers and promoters, from snATAC-seq data that are robust to technical batch effects. Enhlink parsimoniously identifies links between enhancer genomic regions, identified by peaks, and target

genomic regions such as promoters (but not limited to) across genome-wide candidates. It can also prioritize associations supported by paired (multi-omic) expression data, when available (Fig. 1C, D, and S2). For simplicity, throughout the manuscript, we refer to these genomic features as "enhancers" and the target regions as "promoters." Enlink identifies OCRs and biological factors that "explain" a promoter, independent of technical factors (see "Methods"). It does so by using single-cell representations of features—promoters, enhancers, and biological and technical factors—where each is a binary vector with an element corresponding to the accessibility (or factor label) of each cell. The candidate set of enhancers may be limited to a genomic range surrounding the promoter ($+/-250$ kb, by default) to approximate the promoter's topologically associating domain (TAD)—i.e., a three-dimensional subregion of the genome that sequesters self-interacting regions [22], or may include all peaks genome-wide to model distal [1] and indirect links. Biological and technical factors such as batch, lineage, and genotype and categorical factors can be represented in full generality through "one-hot encoding." Enhlink uses a binary decision tree to iteratively select features that maximize a modified information gain (see "Methods"). It computes an ensemble of such trees, bootstrapping the cells and selecting a random subset of features in each tree, in a manner similar to random forests. Bootstrapping accounts for heterogeneity across datasets and enables the calculation of *p*-values for each enhancer or biological factor. The depth of each tree is controlled by an intuitive hyperparameter, which effectively sets the expected number of enhancers per promoter (four, by default). This depth and random feature subsetting prioritize a reduced set of enhancers (or biological factors) having the strongest, independent association with the promoter.

Enhlink can further prioritize these snATAC-seq-derived enhancers by integrating mRNA measurements simultaneously assayed along with chromatin accessibility on each cell [23]. When such data are available, Enhlink identifies enhancers using both the promoter accessibility profiles and their associated gene expressions. By retaining enhancers that are concordant in both modalities, Enhlink enhances the likelihood of association between the identified enhancers and their target genes (Fig. 1D). Enhlink then refines the snATAC-seq-derived set of promoter-associated enhancers by intersecting them with those derived from snRNA-seq.

Finally, Enhlink can identify enhancers active in a context-specific manner, e.g., those associated with a promoter in a specific biological condition or those cooperating with another promoter-linked enhancer. This is done via a second-order analysis in which the intersection (i.e., product of binary vectors) of a promoter and a biological factor (in the first case) or of a promoter and enhancer (in the second) are substituted for the accessibility profile of a promoter in the above framework (Fig. 1C).

Thanks to the modified information gain (see "Methods"), Enhlink is capable of determining the correlation direction (either positive or negative) of the inferred links. In all the following analyses, we restricted Enhlink (see Table S1), focusing on identifying potential enhancers rather than repressors or insulators. In addition, we computed the ratio of negatively correlated links relative to the total number of links across the nine cell populations investigated in the human heart study mentioned earlier. Our findings revealed that the proportion of negatively correlated links varied from 4 to 15% among different cell populations (Figure S3).

Enhlink implementation and speed

Enhlink achieves its computational efficiency through its implementation in Go (https://go.dev/), a programming language optimized for CPU and memory usage [24]. It computes each decision tree within a distinct thread, allowing computational speed to scale with the number of threads used (Fig. 1E). Additionally, it can distribute the computation of a set of promoters or a grid of hyperparameters over multiple "processes", improving the computational time while preserving the amount of memory needed on high-performance computing (HPC) clusters. Enhlink processed 200 promoters over 100,720 cells in 55 s using ten threads in each of four processes on a cluster of 52 CPUs and in 167 s with 2 threads in each of 4 processes (Fig. 1E). Analysis of a single promoter using all 295,089 genome-wide peaks took approximately 700 s (Fig. 1E). These processing times could have been further reduced with little impact on performance by randomly downsampling cells or peaks, as described in the next section. Memory usage only increased marginally with the number of threads and processes. Overall, Enhlink's execution time for each promoter is linearly proportional to the number of trees in the ensemble, the number of enhancer peaks considered as features, and the number of cells, while it is exponentially dependent on the depth of each tree, and inversely proportional to the total number of threads used (across all processes). We compared the processing time of Enhlink with the processing time of our own parallelized implementation of a Chi2 procedure (see "Methods"), written in Python. The computational speed of a Chi2 procedure depends only on the size of the contingency table, thus making it in theory much faster to compute. However, we noticed that the procedure became slower than Enhlink when using a higher number of threads (Figure S4), highlighting higher overhead in the Python implementation and emphasizing the importance of the software technology used. Enhlink takes as input sparse matrices in an MTX format compatible with Cell Ranger and easily generated from Python and R workflows. Enhlink open-source code is freely available and accompanied by in-depth tutorials (https://gitlab.com/Grouumf/enhlinktools).

Estimating accuracy and power analysis from simulated data

Inspired by the signal detected from snATAC-seq data for observed enhancer–promoter interactions (Fig. 1B), we designed a strategy to simulate enhancer–promoter co-accessibilities with characteristics similar to those previously validated or well characterized. Briefly, we simulated a promoter accessibility vector by randomly shuffling one of the *MYL2* or *KCNH2* promoters observed within vCMs or aCMs and described above. We then simulated enhancer vectors by introducing random noise into the simulated promoter vector (Fig. 2A). To model noise in the simulated enhancers, we defined the probability of a cell having a read at the promoter and not at the enhancer following a Poisson distribution (λ_{close}) or, conversely, at the enhancer and not at the promoter (λ_{open}; see "Methods"). We estimated λ_{close} and λ_{open} from the observed, cell type-specific *MYL2* and *KCNH2* enhancers and found similar values for λ_{close} ($1.9 +/- 0.2$) and λ_{open} ($0.08 +/- 0.3$) (Fig. 2B). We extended this analysis using all the cells and found slightly higher λ_{close} ($2.0 +/- 0.2$) and lower λ_{open} ($0.02 +/- 0.05$) values. In addition, we also found that the values of λ_{close} and λ_{open} were overall robust to downsampling of cells and to read dropouts (Figure S5AB). Finally, λ_{close} and λ_{open} were stable to the

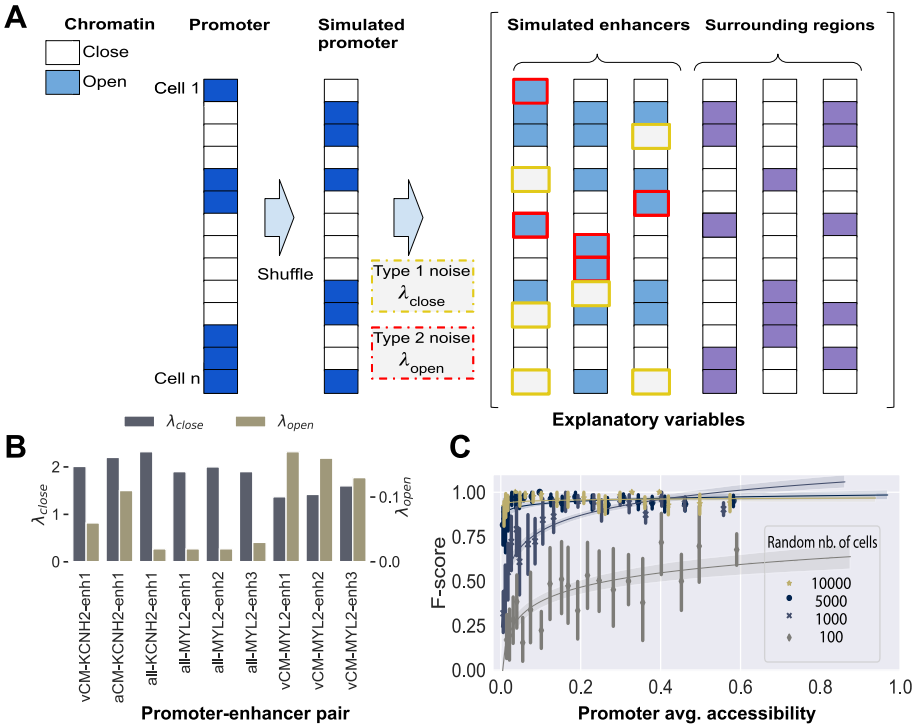

Fig. 2 Empirically parameterized simulation demonstrates Enhlink's high accuracy. **A** Workflow to simulate promoter–enhancer associations parameterized by experimental data. The accessibilities of a promoter and its associated enhancers across cells are simulated from a single promoter–enhancer pair having a validated association. The simulated promoter accessibilities are derived by randomly shuffling the binary, scATAC-seq-derived accessibilities of the validated promoter across cells. Each simulated enhancer accessibility for a given cell is generated from the simulated promoter accessibility for that cell via a process that probabilistically flips the cell's chromatin state: from closed to open (parameterized by λ_{open}) or from open to closed (λ_{close}). λ_{open} and λ_{close} are determined from the validated promoter–enhancer pair. The simulated enhancers are then integrated with the surrounding regions used as background. **B** λ_{open} and λ_{close} distribution parameters inferred from chromatin accessibility of enhancer–promoter pairs previously validated in human scATAC-seq cardiomyocyte cells (Hocker et. al 2021). Pairs involve the promoter *KCNH2* or *MYL2* as determined in all cells or in the subset of aCM or vCM cells. **C** f1-score (*y* axis) of simulated promoter–enhancer pairs as a function of average promoter accessibility and number of cells. Error bars summarize 20 simulated promoters. Each simulated promoter has between two and seven associated simulated enhancers

mixing of a rare or unknown cell type as contamination within a larger population, as would occur following imprecise clustering (Figure S5CD). We then estimated Enhlink precision, recall, and f1-score using the simulated enhancer–promoter associations as ground truth. Simulation across three datasets, for a large number of promoters, and with multiple λ_{close} and λ_{open} parameters (see "Methods") highlighted that accuracy was mostly dependent on average promoter accessibility across cells and number of cells in the dataset (Fig. 2C and S6). Most importantly, it underscored the very high accuracy of Enhlink (f1-score > 0.8) when enough cells were used (> 5000) or when a promoter was widely accessible (average accessibility > 0.2; Fig. 2C).

Enhlink outperforms other methods on simulated datasets

We compared Enhlink performance with that of other popular co-accessibility approaches implemented in Cicero [9, 16], SnapATAC [16], Signac [14], ArchR [15], and Robustlink [17] (Fig. 3A). We also performed a contingency table analysis relating

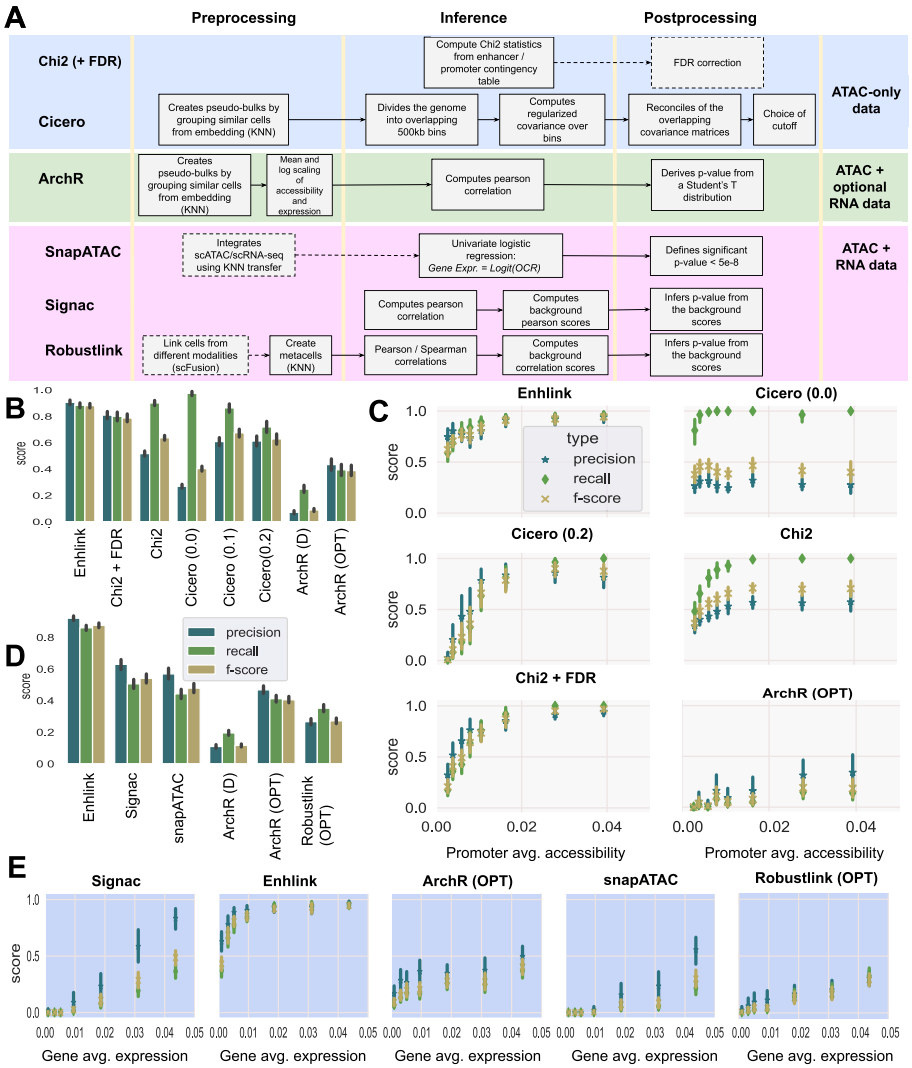

Fig. 3 Enhlink outperforms other strategies for inferring linkage on simulated data. **A** Summary of existing enhancer–promoter method workflows. Some methods use scATAC-seq only as input (Cicero, Chi2 + FDR), others use scATAC-seq combined with scRNA-seq (Signac, SnapATAC, Robustlink). ArchR has a mechanism for both cases. **B** Enhlink outperforms ATAC-only methods on 400 simulated promoters and 1800 simulated enhancers generated from scATAC-seq data. The scores are computed from the average performance from each simulated promoter (see "Methods"). (OPT) refers to the selection of optimal hyperparameters for ArchR and (D) for the default values. **C** Enhlink outperforms other ATAC-only methods independently of the promoter accessibility. Accuracy is dependent on the promoter accessibility (*x* axis) with more accessible promoters leading to better f1-scores. **D** Enhlink outperforms ATAC + RNA methods on 897 simulated genes and 4090 simulated enhancers inferred from the multiome snRNA-/snATAC-seq data. Robustlink (OPT) is obtained with a resolution of 50.0 **E** Enhlink outperforms other ATAC + RNA methods across average gene expression values. Accuracy is dependent on the gene expression (*x* axis) with more expressed genes leading to better f1-scores (*y* axis)

a promoter and enhancer (Chi2) and, optionally, corrected the resulting *p*-value for genome-wide multiple hypothesis testing using the method of Benjamini and Hochberg [25] (Chi2 + FDR). Rather than utilizing the R implementation of the linkage inference workflow from ArchR, Signac, and SnapATAC, which are embedded within larger processing workflows, we opted to re-implement the algorithms in a Python framework (see

"Methods"). This decision significantly streamlined the process of inference across different methods using the same datasets. Cicero and the Chi2 approaches are applicable to snATAC-seq data only, while Robustlink [17], SnapATAC [16], and Signac [14, 16] were applied to correlate enhancers with gene expression. ArchR was applied to both snATAC-seq data only and to correlate genes (continuous values) with enhancers (binary). Also, rather than using the default ArchR hyperparameter values, we tested a grid of *knn* and *n* hyperparameter values (see "Methods") to obtain the highest performance (Figure S7). Similarly, we also selected the optimal resolution for the Leiden algorithm leading to the highest performance for Robustlink (Figure S8). We simulated one dataset of promoter and associated enhancer accessibilities, using the framework described above, and a second dataset of gene expression and associated enhancer accessibilities, using a similar framework (see "Methods"). Both datasets were derived from cell type (i.e., aCM or vCM)-specific λ_{close} and λ_{open} parameters and, thus, effectively simulate cells of a single cell type. We applied each method to the same set of query promoters and enhancers (see "Methods"). Enhlink outperformed other methods in terms of f1-score and precision computed for each gene/promoter (Fig. 3B-C), with Chi2 + FDR and Cicero (with a score cutoff of 0.2) matching Enhlink performance only for more accessible promoters (average accessibility > 0.025; Fig. 3D). In the experiment with simulated gene expression, Enhlink performance (f1-scores ~ 0.88) greatly exceeded those of Robustlink, Signac, and SnapATAC (f1-scores ~ 0.50) (Fig. 3E). We considered modifications to ArchR, including changing parameters for the embedding step and replacing its *p*-value calculation with the one used by Signac. The results showed that using fewer neighbors (see "Methods") increased the accuracy of ArchR with simulated genes (Figure S7) and suggested that the first embedding step of ArchR was actually detrimental to its accuracy.

Enhlink enhancer–promoter associations are enriched for physical interactions

We next applied Enhlink to scATAC-seq and (promoter capture) PCHi-C data generated across two tissues—pancreatic islets and adipose—in our previous T2D study (Poirion et al., 2024). This study examined the effect of mouse genotype (i.e., strain), sex, and diet on the cellular heterogeneity of these metabolic tissues in mice fed an obesogenic or laboratory diet. In both tissues, accessibility profiles clearly separated major cell types (Fig. 4A and C), and we previously reported differences between genotypes and diets (Poirion et al., 2024). Here, we applied Enhlink, Chi2 + FDR, and Cicero independently to each tissue and cell type to identify OCRs co-accessible with promoters. PCHi-C was previously performed on these same tissues to identify physical enhancer–promoter interactions (see "Availability of data and materials"). In both tissues, we observed higher recalls, precisions, and f1-scores (see "Methods") for Enhlink compared to Cicero, and much higher numbers of links leading to higher recalls but lower precision for Chi2 + FDR compared to Enhlink (Fig. 4B and D). Here, we did not apply a cutoff to filter Cicero links, so as to obtain a number of links similar to that resulting from Enhlink. Further, we found that Enhlink-inferred links were more likely to be shared across sequencing batches (as measured by entropy computed across batches, see "Methods") and hence less likely to be induced by technical artifacts, than were those inferred by Cicero or Chi2 + FDR (Fig. 4F). Indeed, the proportion of links with *zero* or close to zero entropy, indicating a batch-specific link and a possible batch effect, was lower among

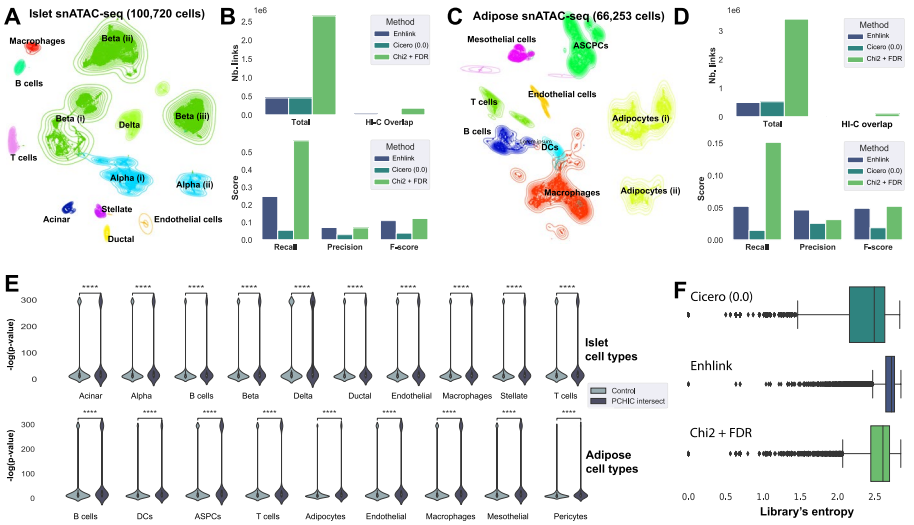

Fig. 4 Enhlink outperforms other approaches in retrieving PCHi-C links and mitigates batch effects. **A** UMAP embedding and cell types of the islet dataset. **B** Enlink, Cicero, and Chi2 performance of promoter-enhancer inference in islet snATAC-seq relative to islet PCHi-C. **C** UMAP embedding and cell types of the adipose dataset. **D** Enlink, Cicero, and Chi2 performance of promoter–enhancer inference in adipose snATAC-seq relative to adipose PCHi-C. **E** Comparison (Mann–Whitney test) of the Enhlink *p*-value distributions from links intersecting PCHi-C and those not intersecting (control). Levels used for Mann–Whitney *p*-values are **** for *p*-value < 1e-4, *** for *p*-value < 1e-3, ** for *p*-value < 1e-2, and * for *p*-value < 0.05. **F** Distribution of the *batch x link* entropy for Cicero, Chi2, and Enhlink from a subset of cells from the islet dataset. Low entropy close to zero indicates links that exist only in a few or a single batch while high entropy indicates links widespread among the batches

those inferred from Enhlink relative to the other two methods (Figure S9). Finally, we found that Enhlink-inferred links present in the PCHi-C data had lower *p*-values (Mann–Whitney tests comparing the -log(*p*-value) distributions with a 0.05 threshold) than those that did not across all cell types (Fig. 4E). This suggests that Enhlink *p*-values are a good indicator of the biological meaningfulness of a given link. As such, we generated an atlas of enhancer–promoter co-accessibilities for both tissues and each cell type (Figure S10), indicating genotype, sex, and diet-specific effects and annotated these links with PCHi-C interactions (see "Availability of data and materials").

To further investigate the enrichment results concerning reference linkages, we expanded our analysis by incorporating six reference datasets sourced from the EnhancerAtlas 2.0 database [26]. Specifically, we utilized Enhlink, Cicero, and the Chi2 + FDR procedure to infer linkages across the nine cell populations of the CARE dataset, which were previously employed for modeling the enhancers of KCNH2 (Fig. 1). Subsequently, we calculated an enrichment score (see "Methods") based on the reference enhancers and links obtained from EnhancerAtlas for each cell population. Furthermore, we obtained reference datasets from EnhancerAtlas for the mouse islet and striatum and similarly computed enrichment scores for the relevant cell populations using different methods. Our analysis revealed that Enhlink consistently demonstrated superior enrichments compared to Cicero and the Chi2 + FDR method (Figure S11), thus corroborating the findings observed with the reference PCHi-C datasets.

Prioritizing neuronal enhancer–promoter links through multi-omic integration

To characterize epigenomic regulation within another heterogeneous tissue, we generated a multi-omic snATAC-/snRNA-seq dataset from the striatum region of the brain. Striata were collected from eight genetically diverse inbred mouse strains, representing the founders of the Diversity Outbred (DO) and Collaborative Cross populations used for genetic analysis of complex traits [27]. Of particular interest in the striatum is a subset of striatonigral and striatopallidal neurons that are defined by expressing either the dopamine receptor 1 (*Drd1*) or 2 (*Drd2*), respectively [28]. Clustering of gene expression identified eight major cell types and mixed populations of neurons, including the *Drd1*/*Drd2* neurons and all other previously identified major cell types (Figure S12A).

We sought to prioritize links between enhancers and promoters/genes based on both chromatin accessibility and gene expression data. Focusing first on *Drd1* neurons, we used Enhlink to identify 47,682 enhancer–promoter links within scATAC-seq data and 44,101 enhancer-gene links within scATAC-seq and scRNA-seq data. A subset of 16,431 links were concordant (i.e., shared and in the same direction), and these had higher scores and lower p-values than those uniquely identified from co-accessibility alone (snATAC-seq only; Mann–Whitney p-value < 1e-64; Figure S12B-C). Similarly, enhancers within *Drd2* neurons with concordant ($n = 17{,}098$) associations with a promoter's accessibility ($n = 45{,}424$) and its gene's expression ($n = 47{,}023$) had higher scores and lower p-values than those supported by co-accessibility alone (Figure S12B-C). Collectively, these results suggest that joint multi-omic analysis further refines enhancer identification.

We next used the expected association between a gene's expression and its enhancers' accessibility in the multi-omic data to evaluate computationally inferred co-accessibility between the gene's promoter and those enhancers. For each marker gene of the Drd1 or Drd2 neurons, we inferred enhancer links with the associated promoter. We then evaluated the links according to whether they exhibited the expected correlation with the gene's expression, as assessed with logistic regression (see "Methods"). Both Cicero (without cutoff) and Enhlink identified enhancer associations for 162 of the 172 marker genes. However, Enhlink identified a more focused set of links ($n = 802$) relative to Cicero ($n = 6997$), having significantly stronger associations with gene expression (Mann–Whitney p-value < 0.05) (Figure S13). This suggests Enhlink associations are enriched for true positives. Increasing Cicero's cutoff resulted in smaller sets of links having increased correlation with gene expression relative to results without a cutoff, yet still inferior or similar to correlation inferred by Enhlink (Mann–Whitney p-value < 0.05). Further, this came at the expense of *fewer* linked genes—92 for the often used [10, 21] cutoff of 0.2 and 55 for the cutoff of 0.28 yielding the number of links ($n = 700$) most similar to those obtained with Enhlink. Overall, these results show the strength of Enhlink in identifying putative enhancers more strongly associated with target gene expression relative to those identified by Cicero.

Validating neuronal enhancer–promoter links with eQTL

One method to validate putative enhancers is through direct genome editing to test the downstream effect on gene expression. As an alternative approach, genetic variation among individuals provides a natural source of variation within enhancer sequences.

The Diversity Outbred mouse population segregates greater than 50 million variants with extremely high precision for genetic mapping [27, 29, 30]. To explore the ability of Enhlink to identify biologically-relevant enhancers, we integrated previously collected (see "Methods") expression quantitative trait loci (eQTL) data from bulk striatum RNA-seq experiments with enhancer links identified here. Genes with local (or *cis*-) acting eQTL are abundant in the DO population. Further, these *cis*-eQTLs are likely driven by variants within an enhancer proximal to the regulated gene that impacts expression [10], which we hypothesize would alter cell-type specific OCRs identified through snATAC-seq. Because the snATAC-seq data collected here represent replicates from the eight parental strains of the DO, we can further estimate OCR accessibility within each strain and see which accessibility patterns match the eQTL results. We identified 1731 and 429 links from joint analysis of the snRNA- and snATAC-seq that were significantly (Enhlink p-value < 0.01) associated with Drd1 or Drd2 in both modalities, respectively (see "Methods"). To look for enhancers that drive differential expression between these two subclasses of neurons, we compared these links with the set of marker genes from Drd1/Drd2 neuron gene expression clusters (see "Methods"). Doing so identified 159 links for 66 genes for *Drd1* neurons and 32 links for 17 genes for *Drd2* neurons. Of the enhancers identified with Enhlink and associated with marker genes of *Drd1* or *Drd2* neurons, 68 are linked to genes with *cis*-acting eQTL. We performed a SNP association analysis using a logarithm of the odds (LOD) regression approach to identify variants that show an association between the genotype at the OCR and gene expression (see "Methods"). Candidate enhancers driving variation in expression were identified as those with matching correlations between their genotype at the variant, gene expression, and accessibility of the promoter and enhancers across inbred strains. Of the identified correlations, three enhancer–promoter links for marker genes *Gulp1*, *Kcnb2*, and *Col25a1* (Fig. 5A-B) serve as proof of principle. In each case, the strains with the genotype identified as having the largest effect on the eQTL from bulk data (alternative genotype, Fig. 5C) presented differential accessibility and expression in the single-nuclei data compared to the other strains (Fig. 5B). Together, these data show that Enhlink identifies biologically relevant enhancers that play an active role in cell-type- and strain-specific gene regulation.

Identifying distal Drd1-specific enhancers

While the above analysis focused on identifying how local genetic variation impacts gene expression, Enhlink can also be used to identify cell type-specific *distal* enhancer networks. To do so, we extended our analysis by finding distal common enhancers for the top 10 *Drd1*-specific upregulated marker genes (*Adarb2, Cntnap3, Lingo2, Drd1, Il1rapl2, Cntnap5c, Erbb4, Nrg3, Tac1, Ebf1*) and top 10 *Drd2* specific marker genes (*Drd2, Nell1, Necab1, Unc5d, Grik3, Ptprm, Fam155a, Chrm3, Penk, Adk*) across *all* (259,720) OCRs, rather than those within +/− 250 kb of their respective promoters. We identified 110 links from snATAC-seq data alone. To mitigate potential false positives over this genome-wide set of OCRs, we further applied Enhlink to the multi-omic snATAC-/snRNA-seq data as above and inferred 70 links. We prioritized the 33 links shared across both analyses for further analysis. Remarkably, 22 of these 33 links were associated with just four enhancers, all arising from a region 500 kb downstream of the *Drd1* promoter and surrounded by the predicted pseudo-genes: *Gm34439*, *Gm40954*,

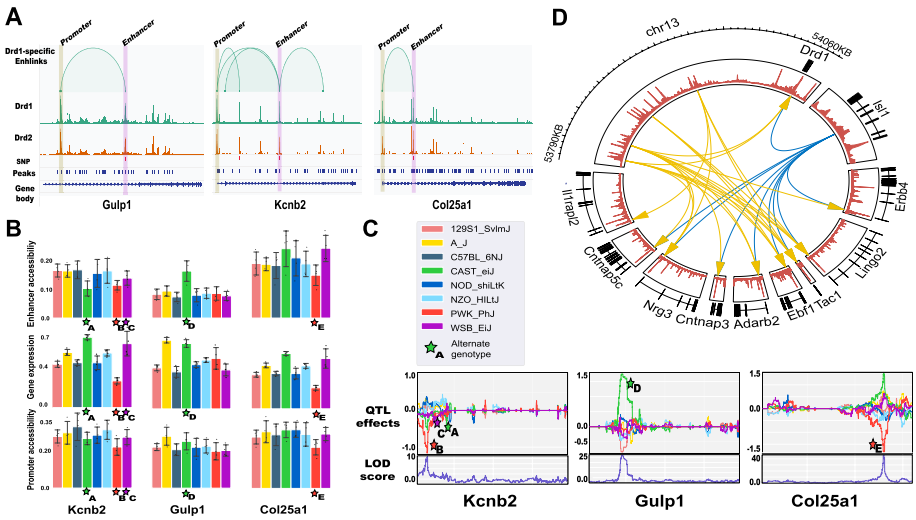

Fig. 5 Enhlink reveals chromatin regulation mechanisms of striatum Drd1/Drd2 neurons. **A** Chromatin accessibility (*y* axis) with Enhlink-inferred links between the promoters and enhancers for *Kcnb2*, *Gulp1*, and *Col25a1*, three marker genes of Drd1 neurons. **B** Chromatin accessibility and gene expression profiles per genotype for three enhancers (*Kcnb2*, *Gulp1*, and *Col25a1*). **C** eQTL logarithm of odds (LOD) scores for SNPs within the boundaries of the three enhancers across the eight DO genotypes. Stars indicate genotype harboring an alternative allele within an enhancer of *Kcnb2*, *Gulp1*, or *Col25a1*. Star subscript associates LOD scores in panel C with chromatin accessibility and gene expression in panel B. **D** Distal Enhlink analysis unveils multiple enhancers from the region 500 kb downstream of the *Drd1* promoter and linked to the top 10 marker genes of *Drd1* neurons (yellow arrows). These genes are also linked to an intronic region of *Isl1* (blue arrows), a key gene regulating Drd1/Drd2 genetic programs

and *Gm34557* (Fig. 5D and **S14A**). Further, all 10 of the *Drd1*-specific genes have at least a distal enhancer within the four *Drd1* proximal enhancers (Fig. 5D and S14A). Additionally, 9 out of the remaining 11 links matched an intergenic OCR from the Islet-1 (*Isl1*) gene that was previously characterized for regulating striatonigral and striatopallidal genetic programs [31] (Fig. 5E and S14B). Notably, neurons from an *Isl1* knockout mouse showed an increase in *Drd2* expression and promoted striatopallidal (*Drd2*) neuron differentiation while repressing striatopallidal (*Drd1*) genes. Together, these data show that Enhlink can identify coordinated chromatin regulation at distal loci with biologically meaningful connections, perhaps indicating coordinated transcription factor activity [32]. All the linkages obtained from Striatum analysis are publicly available (see "Availability of data and materials").

Discussion

In this study, we introduce Enhlink, a novel computational method that efficiently infers genomic linkages from single-cell datasets and is suitable for complex experimental designs. Enhlink infers enhancer–promoter co-accessibility from chromatin accessibility profiles in scATAC-seq data. It can also infer enhancer–promoter links supported by both their co-accessibility and concordant enhancer accessibility and target gene expression within multi-omic snATAC-/snRNA-seq datasets. More generally, Enhlink could be applied to other single-cell modalities containing sparse high-dimensional data, such as

single-cell DNA methylation [33], single-cell ChIP-seq [34], or multi-omic datasets combining epigenome, methylome, and/or transcriptome [35].

Enhlink leverages an original procedure derived from random forests [36] to extend the capabilities of existing methods. First, Enhlink adjusts for technical covariates, such as sequencing library ID, to minimize batch effects at the level of *individual* cells. Methods that instead apply batch correction to aggregates of similar cells do not readily accommodate technical effects *within* the aggregate. This would pose a problem in studies, such as our T2D study, where cells of similar genotype are partitioned across batches (see "Methods"). Second, Enhlink can infer enhancers linked to a promoter within specific contexts, such as sex, genotype, or disease, by including each as a biological covariate. Third, for each enhancer–promoter link, Enhlink infers a *p*-value that is adjusted for multiple hypothesis testing *relative to all and only those enhancers tested for association with that promoter*. In this way, the strength (i.e., adjusted *p*-value) of each inferred association is scaled according to the genomic context of each promoter. Hence, two enhancers may have different inferred associations, even if their correlations with their respective promoters are similar, for example, if one enhancer has a much higher correlation with other enhancers it is compared against than the other enhancer. Fourth, Enhlink can infer linkages from distal regions beyond the neighboring OCRs. Finally, Enhlink can perform power analyses to estimate expected accuracies based on a simulation workflow developed from experimentally validated enhancer–promoter linkages. This simulation framework can be applied independently of Enhlink and can aid others in diagnosing the impact of hyperparameter tuning, preprocessing steps, or other methodological choices.

Enhlink employs several regularization mechanisms to reduce false positives. One hyperparameter, the maximum number of explanatory features of each tree, is biologically interpretable as the *expected* number of enhancers at each target region. While the actual number of enhancers is a property of the entire *ensemble* of trees, we expect that it is of the same order of magnitude as the number of features considered at each *tree*. As such, the hyperparameter can be set according to biological expectations or tractability of downstream experimental validation. Enhlink sensitivity and specificity can also be fine-tuned by adjusting the number of trees used and the minimum number of features required for each tree. Thus, weak but significant enhancer–promoter associations could be inferred by using a larger forest and feature size at the expense of increased computation. We implemented Enhlink in Go enabling it to be extensively distributed among a cluster of CPUs and providing superior resource management compared to many computational single-cell tools written in R or Python. For example, for a large number of threads Enhlink became faster than our parallelized implementation of the Chi2 + FDR procedure, written in Python, indicating better overhead and CPU usage. This makes Enhlink well-suited for analyzing current [37] and future [23] large-scale multi-omic single-cell datasets.

Through extensive benchmarking of both simulated and real data, our study demonstrates that Enhlink significantly outperforms other existing approaches in terms of accuracy while also limiting artifactual, batch-specific linkages. We took advantage of multi-omic data to show that enhancers identified from scATAC-seq were better correlated with gene expression when inferred with Enhlink than with the popular Cicero

framework. In addition, simulation and intersection with two reference PCHi-C datasets showed that Cicero was also outperformed by a Chi2 procedure followed by FDR correction. We also showed that aggregating cells into pseudo-bulk, a preprocessing step followed by ArchR and Robustlink and inspired by the Cicero workflow, was actually detrimental to accuracy. This occurs when the binning, aimed to group cells with similar characteristics, is too coarse and results in the grouping of heterogeneous cells. We evaluated Enhlink using three datasets, including two snATAC-seq datasets previously generated from mouse islet and adipose tissues and a novel multi-omic snATAC-/snRNA-seq dataset from the mouse striatum. The single-cell islet and adipose studies aimed to characterize cellular heterogeneity in these tissues and uncover genes and regulatory elements perturbed by diet and genetic mechanisms. To aid this investigation, we developed cell type-specific enhancer–promoter atlases of the islet and adipose tissues (see "Availability of data and materials"). These include genotype-, diet-, and sex-specific linkages that are annotated according to whether the corresponding promoter and enhancer physically interact based on PCHi-C data generated from the same tissues. Links supported by PCHi-C data, and those with concordant promoter chromatin accessibility and target gene expression in the striatum dataset, have significantly lower Enhlink *p*-values than their counterparts. This suggests that Enhlink *p*-values robustly reflect the biological meaningfulness of their associated links.

We generated the multi-omic mouse striatum dataset to investigate epigenomic regulations underlying the differentiation and gene regulation of striatonigral neurons expressing the dopamine receptor *Drd1* and striatopallidal neurons expressing *Drd2*. Our goal was to identify strong candidate regulatory regions involved in these processes. To achieve this, we utilized Enhlink to identify concordant links between promoter accessibility and gene expression. We then examined the relationship between naturally occurring regulatory variation, chromatin accessibility, and gene expression among eight parental haplotypes with associated eQTL and SNP data. Through this analysis, we identified three enhancers associated with the expression of *Gulp1*, *Col25a1*, and *Kcnb2*. The contribution of the different haplotypes to the expression pattern indicated that the variants located within the Enhlink-identified enhancers were likely to be the causal factors for the observed changes in accessibility and expression. This finding strengthens the hypothesis that the regions identified by Enhlink play a crucial role as enhancers. Given that the majority of disease-associated variants within the human population occur within OCRs, this approach is extendable to prioritizing variants related to complex traits and might palliate some of the theoretical limits of eQTL analysis [38].

We further demonstrated the power of Enhlink to detect links between a target promoter and its distal enhancers ($>+/-250$ kb). We hypothesized that key enhancers should be detected as hubs correlating with the expression of multiple genes, similar to other studies modelling gene expression with SNPs [39, 40]. Strikingly, we found that most of the distal links inferred from the top marker genes of the striatonigral neurons came from a region located 500 kb downstream of the *Drd1* promoter. This region exhibits many characteristics of a super-enhancer region [2], as it contains a cluster of enhancers associated with several marker genes of the striatonigral neurons. Super-enhancers have been shown to determine cell fate and to maintain cell identity [41],

stressing the possible role played by the downstream region of *Drd1*. Also, we found that all top ten markers of *Drd1* neurons were linked with an intronic region of the *Isl1* gene, a key gene in the striatopallidal/striatonigral differentiation [31]. While these results are novel, they need to be confirmed through further experimental validation.

While inferring Drd1 and Drd2 linkages we noticed that only ~ 35% of the links were shared between the ATAC and ATAC + RNA links. We attribute these disparities to two primary factors. Firstly, accessible promoters are not always expressed. Secondly, transcription can occur in bursts [42] producing either persistent or short-lived [43] mRNA molecules, while the epigenomic profiles of promoters follow their own dynamic patterns. Consequently, we believe that linkages derived from ATAC data alone hold greater relevance than those from ATAC + RNA data and should therefore be prioritized. Additionally, identifying concordant sets of links within a multi-omic dataset may enhance linkage specificity but could potentially reduce sensitivity.

Enhlink is a powerful and robust method for inferring genomic linkages from sparse, high-dimensional, single-cell omic, or multi-omic datasets. Enhlink outperforms other tested approaches, in part, by introducing cellular-level covariates that ameliorate technical effects and capture biological effects. Enhlink's efficient implementation is tailored to large-scale single-cell analyses, including those aimed at deciphering complex regulatory networks.

Methods

Human heart snATAC-seq dataset processing

We downloaded the Human Heart snATAC-seq dataset from the portal (http://ns104190.ip-147-135-44.us/CARE_portal/ATAC_data_and_download.html) described in the publication [21]. From the portal, we downloaded the matrix file (*all.npz*), the cell index file (*all.index*), the OCR features index file (*all.merged.ygi*), the genome reference (Homo_sapiens.GRCh38.99.TSS.2 K.bed), the cluster file (*all.cluster*), and the *all.group* file with the library ID of each cell. The matrix from the *all.npz* file is a scipy sparse matrix with *79,515* cells and *287,415* OCRs with boolean values indicating if at least one read mapped to the cell is found within the boundaries of the corresponding OCR. From the genome reference and features index files, we defined the *KCNH2* promoter regions by the following OCRs: "chr7:150,976,584–150977120", "chr7:150,978,193–150978805", and "chr7:150,978,915–150979661". The *KCNH2* enhancer was defined as the following genomic region: "chr7:150,955,147–150956502". We also defined the *MYL2* promoter regions by "chr12:110,920,282–110920944" and "chr12:110,921,386–110921633". We defined three enhancer regions of *MYL2* by "chr12:110,931,149–110931877", "chr12:110,928,658–110929096", and "chr12:110,907,461–110909456". We then focused on the atrial (aCM) and ventricular (vCM) cardiomyocyte cell types, since *KCNH2* is only expressed in these cell types, and defined a *KCNH2* promoter boolean vector for either aCM or vCM by merging (using a *logical or* operand) the three promoter region vectors from the feature matrix using either the cell index of aCM or vCM. We followed the same strategy for the promoter of *MYL2*, but restricted to the vCM cell type since *MYL2* is only expressed in vCM. We also downloaded the bigwig tracks from the same portal for the aCM and vCM cell types from the same portal. Finally, we computed the

Enhlink co-accessibilities for these two promoters and for aCM and vCM based on the workflow described below.

Co-accessibility signals from KCNH2 and MYL2

We used the *KCNH2* promoter vectors of aCM and vCM as ground truth labels (either accessible or not for a given cell) and the *KCNH2* enhancer vectors as estimated labels to compute the precision, recall, and f1-scores. We then drew a random subset of 500 enhancers among the 287,415 features of the feature matrix to compute the baseline distributions for the precision, recall, and f1-score. We then derived a p-value for the computed precision, recall, and f1-score with regard to the baseline distributions. We followed the same strategy for the *MYL2* promoter and its associated enhancers.

Mouse islet and adipose scATAC-seq

We used two scATAC-seq datasets from the mouse islet and adipose tissues whose generation and processing we previously described (Poirion et al., 2024). The islet dataset gathered 100,720 cells, 295,089 OCR features, 10 cell populations including alpha, beta, and delta cells, and was processed using 18 10X Genomics sequencing libraries. The adipose dataset gathered 60,229 cells, 311,645 OCR features, 9 cell populations including adipocytes and macrophages, and was processed using 24 10X Genomics sequencing libraries. Cells from both datasets were labeled with their mouse strain/genotype ID (CAST, NZO, B6), diet (4% or 44% fat), and sex (M/F).

Mouse (RNA/ATAC) multi-omic single-cell collection

Mouse striatum was dissected from 12-week-old mice from eight inbred strains: A/J (The Jackson Laboratory Stock 000646), C57BL/6 J (000664), 129S1/SvImJ (002448), NOD/ShiLtJ (001976), NZO/H1LtJ (002105), CAST/EiJ (000928), PWK/PhJ (003715), and WSB/EiJ (001145). Striatum was collected from two males and two females for each strain. Striatum samples were flash-frozen in liquid nitrogen and stored at − 80 °C until processing. After collection, the striatum samples were processed in four batches of eight over 2 days. Each batch consisted of four male and four female samples, and each strain was represented in each batch.

Multi-omic library generation

For single-nuclei preparation, frozen striatum was placed into 500 µl Miltenyi Nuclei Extraction Buffer (Miltenyi 130–128-024) plus 0.8 U/µl RiboLock RNase Inhibitor (ThermoFisher EO0382) in a gentleMACS C tube (Miltenyi 130–093-237) on ice, and then nuclei were immediately extracted in batches of four through the "4C_nuclei_1" program on the MACS Dissociator. Nuclei were placed on ice and filtered through a 70-µm pluriStrainer Mini into a pre-chilled 2-ml tube, and the C tube was washed with an additional 500 µl cold Miltenyi Nuclei Extraction Buffer and filtered. Nuclei were spun down at 350 rcf at 4 °C for 5 min and resuspended in 80 µl cold PBS with 2% BSA and 1 U/µl RiboLock. Nuclei were counted (~ 4000 nuclei/µl) and 60,000 nuclei from each of the eight samples were mixed in a chilled 1.5-ml Eppendorf tube. The pooled nuclei were spun at 300 rcf at 4 °C for 3 min and resuspended in 30 µl 10 × Genomics

Nuclei Buffer with 1 U/μl RNase inhibitor. The pooled nuclei were counted once more to confirm counts (~ 10,000 nuclei/μl), single-cell suspension, and lack of debris.

Nuclei viability was assessed on a LUNA-FX7 automated cell counter (Logos Biosystems), and up to 40,000 nuclei (~ 5,000 from each sample) were loaded onto each of 4 lanes of a 10 × Chromium microfluidic chip. Single nuclei capture and library preparation were performed using the 10 × Chromium platform and according to the manufacturer's protocol (#CG000388 Chromium Next GEM Single Cell Multiome ATAC + Gene Expression). Because the 10 × chip was superloaded, to reduce duplicate reads arising from multiple PCR steps, the number of cycles was adjusted from the 10 × protocol. After GEM generation, barcoded cDNA and transposed DNA fragments were pre-amplified using 6 cycles. The pre-amplified sample was divided and used for two separate steps. An additional 6 cycles were used for adding a sample index for ATAC library construction and 12 PCR cycles for the gene expression library construction. cDNA and ATAC libraries were checked for quality on Agilent 4200 Tapestation and ThermoFisher Qubit Fluorometer, and quantified by KAPA qPCR, before sequencing; each gene expression library was sequenced using NovaSeq 6000 S4 v1.5 200 cycle flow cell lane, dual index scRNAseq asymmetric read configuration 28–10-10–90, targeting 20,000 nuclei with an average sequencing depth of 50,000 read pairs per nucleus. Each ATAC library was sequenced using Illumina NovaSeq 6000 S2 v1.5 100 cycle flow cell lane, with a 50–8-24–49 read configuration, also targeting 20,000 nuclei with an average sequencing depth of 50,000 reads per cell.

Multi-omic library sequencing

Illumina base call files for all libraries were demultiplexed and converted to FASTQ files using bcl2fastq 2.20.0.422 (Illumina). A filtered joint digital gene expression and chromatin accessibility matrix was generated against the 10 × Genomics mm10-2020-A reference build (version 2020-A, Assembly: GRCm38, ENSEMBL release 98; Annotations: Gencode vM23) using a modified 10 × Genomics CellRanger-ARC count pipeline (v2.0.0), which had the 20,000 cell limit of cell calling removed.

scRNA-seq analysis for the striatum dataset

For each batch of the libraries, the output from Cell Ranger ARC consisted of both ATAC and gene expression BAM files. These two BAM files were used as inputs for Demuxlet [44] to determine the mouse strain identities of individual cells and to detect doublets. Only cells that were identified as singlets with consistent mouse strain identities in both assays were retained for further analysis.

To annotate and visualize the retained single cells, we performed downstream gene expression analysis on them using Seurat (version 4.0.3, [45]). For each batch, we removed low-quality cells and multiplets by filtering out cells with fewer than 200 or more than 7500 detected genes and those with greater than 15% mitochondrial counts. We then used DoubletFinder [46] to further exclude any remaining doublets. Next, we merged the Seurat objects from the four batches and performed normalization, highly variable feature identification, scaling, and linear dimensionality reduction using the NormalizeData(), FindVariableFeatures(), ScaleData(), and RunPCA() commands, respectively, with default parameters. To remove batch effects, we integrated the

single-cell datasets from the four batches using Harmony [47]. Using the first 40 principal components determined manually by the Elbow plot, we conducted unsupervised cell clustering through the Louvain algorithm on the K-nearest neighbor (KNN) graph with resolution set to 0.2, which resulted in 22 cell clusters. We visualized the output in a 2D Uniform Manifold Approximation and Projection (UMAP) embedding using the same PCs used for cell clustering. We excluded four cell clusters that (1) co-expressed neuron markers *Drd1/Drd2* and oligodendrocyte marker *Aspa*; (2) co-expressed neuron markers *Drd1/Drd2* and astrocyte marker *Gja1*; (3) co-expressed neuron markers *Drd1/Drd2* and macrophage marker *C1qa*; or (4) highly expressed mitochondrial genes. We then re-processed and integrated the remaining cells using the same steps as before, yielding 18 cell clusters. We annotated these 18 cell clusters using marker genes identified through the FindAllMarkers() command through DropViz [48].

We processed the snATAC-seq data separately from the scRNA-seq data and followed the same preprocessing procedure as the one described for the mouse islet and adipose snATAC-seq dataset (Poirion et al., 2023). Briefly, we aligned the reads to the mm10 genome with Cell Ranger V6 with default parameters. We inferred the peaks fusing MAC2 [49].

Enhlink analytical workflow

The analytical procedure carried out by Enhlink involves multiple steps illustrated in Figure S2 and explained in greater detail in Additional File S1. It can be summarized as follows: (a) create a feature matrix (i.e., OCR × cell matrix) and a response vector (i.e., single-cell promoter accessibility or target gene expression) for each target genomic region, (b) model the response vector as a function of the feature matrix and identify the significant features associated with the target region, (c) optionally perform a secondary analysis to detect biological covariates associated with the linkage, and (d) in the case of multi-omics data, a linkage analysis is conducted for each omic and consensus links, found in all omics, are then inferred.

At a minimum, Enhlink requires a boolean sparse matrix, indicating the OCRs of each cell and a list of genomic regions, typically the promoters, defining the linkage targets. In addition, Enhlink optionally takes a sparse matrix containing the values of the target regions (such as a single-cell expression matrix in the case of multi-omics data), a file containing the covariates of each cell, and the cluster IDs of each cell. Enhlink constructs a feature matrix M_n for each target region by iterating through the OCRs surrounding it ($+/-250$ kb by default). If the feature matrix contains fewer than 100 features by default, Enhlink supplements it with random features derived from the existing features. After constructing the feature matrix for each target region, Enhlink incorporates one-hot-encoded covariates and generates a response vector v_g, representing either the boolean accessibility or the expression of the gene/target region g. If v_g is continuous (e.g., representing gene expression), it is binarized using the mean of the non-null values as threshold. This would be the case when processing multi-omics data for which gene expressions from the RNA-seq are used to create the target regions. Then, Enhlink proceeds to model $v_g = f(M_n)$ for each cluster (or for all cells if no clusters are provided) and identifies features from Mn that significantly predict Vg. Enhlink employs

a strategy similar to that of a random forest classifier (Breiman, 2001), performing 100 iterations by default. In each iteration, a random sample of cells and features is selected, and a decision tree [50] is used to recursively reduce the number of samples and features based on the top feature selected at each level of the tree. The top features are selected using a modified score derived from the Information Gain (IG) (see Additional File S1), which favors informative, positively correlated, and accessible features. For each feature, Enhlink calculates a *p*-value for its score to be different from 0 across iterations using Student's *t*-test from the disturb library (https://pkg.go.dev/gonum.org/v1/gonum/stat/distuv) and corrects the feature *p*-values using the Benjamini–Hochberg false discovery rate (FDR) procedure.

Estimating the expected accuracy of inferred links through simulation

Enhlink provides an option to estimate the expected accuracy of the inferred links for a given target region by generating simulated enhancers and target regions. This simulation procedure aims to replicate the observed correlations between experimentally validated enhancers and promoters. More detailed information on this procedure can be found in Additional File S1. Briefly, Enhlink first simulates a promoter by shuffling a target region. Then, to simulate an associated enhancer, Enhlink duplicates the simulated promoter and introduces two types of random noise. The noise is controlled by two hyperparameters, λ_{open} and λ_{close}, which model the scenario where the enhancer is not accessible in a given cell (λ_{open}) or the target region is not accessible (λ_{close}). The values of λ_{open} and λ_{close} are estimated from the heart snATAC data, using the experimentally validated enhancer of *KCNH2* and enhancers from *MYL2*, as described below.

Inferring biological context-specific enhancer–promoter interactions

Enhlink can optionally infer biological context-specific linkages with the details of the procedure found in Additional File S1 and summarized here. Each context is represented by a cell-level categorical covariate. Note that Enhlink one-hot encodes categorical variables into boolean features and cannot currently process continuous covariates. In this configuration, briefly, Enhlink first infers the set of enhancers and covariates linked to a target region *g* then, for each of these enhancers, *e* Enhlink computes $Veg = Ve \circ Vg$ as the Hadamard product between the target vector *Vg* and the enhancer vector *Ve*. *Veg* corresponds to a boolean vector indicating when both the enhancer *e* and the target region *g* are accessible within a cell. *Veg* is then used as a new target vector to find the covariates significantly associated with it.

Addressing class imbalance in datasets with unequal covariate distributions

Enhlink provides an option to mitigate class imbalance in datasets with varying covariate distributions, which is beneficial when one or more covariates are under- or over-represented. In such cases, the bootstrap samples may not adequately represent the covariate space. Enhlink addresses this issue by generating a more balanced distribution of covariates through an incremental process. Specifically, Enhlink iteratively selects subsets of cells according to their covariates to obtain a near-uniform distribution of the covariates. While this strategy can be helpful in achieving a more uniform distribution of covariates,

it is important to note that it may produce biased results if one or more covariates are vastly underrepresented. Therefore, it is recommended to thoroughly assess the distribution of covariates and to consider alternative approaches such as covariate removal if necessary to ensure more representative results. More details are given in Additional File S1.

Enhlink hyperparameters

The Enhlink inference workflow is governed by multiple hyperparameters summarized in Table S1. The main hyperparameters governing the regularization are *max_features*, which controls the maximum number of features that a tree can use, and *depth* which controls the maximum depth of each tree. *Depth* and *max_features* are intertwined since *depth* also indirectly controls the maximum number of features. However, *max_features* is a more intuitive hyperparameter to use than *depth*. The latter influences the speed of Enhlink and is set to 2 by default, but needs to be increased if *max_features* is set higher. *secondOrderMaxFeat* (set as 2 by default) is the equivalent of the *max_features* parameter for the inference of the biological context-specific enhancer–promoter interactions, triggered with the *secondOrder* hyperparameter. *N_boot* defines the number of trees to build and controls the false positive rate and influences the speed of Enhlink, similarly to *min_matsize*, controlling the minimum number of features of *Mn*. The hyperparameters *downsample* (the size of the bootstrap) and *maxFeatType* (the number of features to include) can be used to increase the speed of the procedure but lower values might lead to lower accuracies. Finally, *threshold* defines the *p*-value cutoff and can be used to regulate the precision and the recall.

Enhlink software suite

Enhlink is an analytical framework developed in Go (https://go.dev/) and compiled into three executables: *enhlink, enhgrid,* and *enhtools*. The command line manual and arguments of each executable can be accessed using the *-h* flag, (e.g., *enhlink -h*). *enhlink* is the main executable that launches the Enhlink pipeline, while *enhgrid* allows launching Enhlink for a range of input values for all the hyperparameters accepting a numerical value. *enhgrid* is useful for, for example, automatizing a grid-search approach by trying a combination of multiple hyperparameters or for testing different noise levels. *enhtools* intersect results from multiple runs and output either the common or unique links of a particular run. It also computes the accuracy between two runs (f1-score, precision, recall). Finally, *enhtools* can filter links that are not within specific regions defined in an input BED file. This functionality can be used for example to filter links not inside topologically associated domains (TAD). The Go source code, the manual, and the tutorial are available here: (https://gitlab.com/Grouumf/enhlinktools).

Simulation studies of the performance impact of hyperparameters

We conducted a simulation study to estimate the impact of the different hyperparameters using three scATAC-seq datasets: the mouse islet, the mouse adipose, and the snATAC-seq from the multi-omic striatum dataset mentioned earlier. We aimed to investigate the impact of various hyperparameters and experimental conditions, such as dataset origin and read depth, on Enhlink's expected accuracy (see Fig. 2C). To

accomplish this, we first subset the datasets by selecting specific cell types, such as beta cells for the islet, adipocytes for the adipose, and Drd1 neurons for the striatum. We then utilized Enhlink's simulation framework, described in Additional file S1, to estimate the expected accuracy of Enhlink analysis at a given promoter. We used a consistent grid of hyperparameters, as detailed in Table S2, for each dataset and applied the enhgrid tool to perform Enhlink on the hyperparameter grid for all three datasets. Furthermore, we conducted the Enhlink analysis on a random subset of 40 genes from the features list of the three datasets for each combination of hyperparameters in the grid.

Estimating the hyperparameters importance from the simulation studies

From the hyperparameters simulation study described above, we estimated the most significant hyperparameters by fitting a Random Forest regressor from the scikit-learn library [51] using the f1-score as the dependent variable and Enhlink's hyperparameters as predictive variables (Figure S6). The simulation generated the true positive rate (TPR), false positive rate (FPR), and false negative rate (FNR) for each gene tested and for each hyperparameters combination. We then computed the f1-score as $f1_{\text{score}} = \frac{\text{TPR}}{(\text{TPR}+0.5.(\text{FPR}+\text{FNR}))}$ and model it as a function of $f1_{\text{score}} = f(\text{hyperparameter})$. We one-hot encoded all the hyperparameters and their values, removing the first value to avoid colinearity, and used them as binary variables. We also added the dataset ID as an additional variable. We then collected the feature importance of the model, computed as the Gini importance, an impurity-based measurement, using the *feature_importances_* attribute of the scikit-learn class. For each simulated promoter, we also computed the accessibility ratio as the ratio between the number of cells having the promoter accessible and the total number of cells. We then plotted the influence of the promoter accessibility ratio on the f1-score using the Python library *seaborn*.

Generating reference datasets for methods comparison

We generated two simulated datasets, using either snATAC-seq only from the mouse islet or the snATAC-seq + scRNA-seq from the striatum dataset, containing either simulated enhancer–promoter co-accessibilities (ATAC-seq only) or correlated enhancer-gene links (ATAC + RNA). From the mouse islet dataset, we used the delta cell subset and a random set of 400 genes to simulate 400 promoters and 1800 enhancers. For each random promoter, we generated a random number of simulated enhancers (between 2 and 7) with the noise parameters $\lambda_{\text{close}} = 1.25$, $\lambda_{\text{open}} = 0.25$. We created dummy genomic coordinates for these enhancers in order for them to be in the vicinity of their matching promoter. These simulated enhancers/promoters were injected in a matrix of 10,100 cells (the delta cells) and 295,089 peaks (the total number of peaks). From the striatum dataset, we used the Drd1 neuron cell types and a random set of 897 genes to simulate between 2 and 7 enhancers for each gene, for a total of 4090 enhancers. We used the same noise parameters: $\lambda_{\text{close}} = 1.25$, $\lambda_{\text{open}} = 0.25$, and binarized the randomized gene vectors using the mean of its non-null element before generating their associated enhancers (see above). The simulated enhancers were further injected into a matrix of 10,000 cells and 259,720 peaks.

Methods comparison procedure

We processed the ATAC simulated matrix with Enhlink, Chi2, Chi2 + FDR, Cicero, and ArchR workflows and processed the ATAC + RNA simulated matrix with Enhlink, Signac, SnapATAC, and ArchR workflows. We used the same genomic window of +/− 250 kb around the promoter for all workflows. We then computed the$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}}$,$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}$, and the $f1_{\text{score}} = 2\frac{\text{Precision . Recall}}{\text{Precision} + \text{Recall}}$ for each of the simulated promoters/genes using the set of simulated enhancers as the true positives. We computed for each workflow and dataset the overall accuracy scores with their standard deviation (sd) using the mean and the sd of the three metrics. Finally, we also plotted the correlation between the accuracy scores and the mean of each simulated gene or promoter using the *lmplot* function of the Python *seaborn* library.

Alternative workflows implementation

Chi2

A straightforward approach for inferring the co-accessibility of a promoter region g with the set E of its surrounding enhancers is to perform a Chi2 test between g and e for each $e \in E$. Since v_g and v_e, i.e., the accessibility vector of g and e, were binary, we constructed a 2×2 contingency table for each (g, e), containing the occurrence for the following conditions: $v_g == 1 \cap v_e == 1$, $v_g == 0 \cap v_e == 1$, $v_g == 0 \cap v_e == 0$, and $v_g == 1 \cap v_e == 0$. We used the *chi2_contingency* function from the Python library *Scipy* [52] to infer the p-value.

Chi2 + FDR

A more refined approach to the Chi2 method described above is to correct for multiple hypothesis testing by applying the false discovery rate procedure from Benjamini–Hochberg [25] on the set of p-values obtained when applying the Chi2 on E. The method consisted in computing an expected p-value (fdr), assuming a false positive, based on the rank of the feature and the corrected p-value was equal to *p-value x fdr*. We used the *fdrcorrection* method from the Python *Statsmodels* library [52] to perform the FDR corrections.

Cicero

We downloaded the latest version of the Cicero algorithm [9] using the *R* library developed by the authors (https://cole-trapnell-lab.github.io/cicero-release/docs/). To parallelize the workflow and reduce the shared memory used, we processed each chromosome independently. For each chromosome, we created a UMAP embedding with the following steps: (i) we performed a TF-IDF embedding using the *TfidfTransformer* class from Scikit-Learn. (ii) We used the singular value decomposition (SVD) using the *TruncatedSVD* class to embed the data into 25 components. (iii) We transformed the 25 components with the Harmony algorithm correcting for the library ID. (iv) We used the UMAP class from the *umap* Python library with the "*correlation*" as a metric, 2.0 as *repulsion_strength*, and 0.01 as *min_dist*. Our Cicero workflow consisted of the following steps: (a) load the sparse matrix and creating a Cicero data object with *make_atac_cds* and *binarize* set as *TRUE*, (b) aggregate the raw count data with *make_cicero_cds* and

$k = 50$, where cell similarity used to bin cells is determined in the harmony-transformed UMAP embedding, (c) estimate the distance parameter with *estimate_distance_parameter* using *window* = *250,000*, *maxit* = *100*, *sample_num* = *100*, and *distance_constraint* = *25,000* and computed the mean of the distance parameters, (d) generate the Cicero models with *generate_cicero_models* using the mean distance parameter and *window* = *250,000*, and (e) assemble the connections with *assemble_connections* and save all the Cicero connections.

Signac

The Signac methodology to identify enhancers significantly correlated with the expression of a given gene was described in the Method *Peak-to-gene* section of the published study [14]. The Signac methodology was described in four steps: (i) compute the Pearson correlation between the gene expression and the accessibility of each peak within 500 kb. (ii) For each peak, compute a background distribution using 200 random peaks matching the GC content, the accessibility, and the sequence length of the peak, and identified with the *MatchRegionStats* R function (https://github.com/stuart-lab/signac/blob/HEAD/R/utilities.R). (iii) Compute a z-test using the z-score obtained with the mean and variance of the background distribution. (iv) Retain links with p-value < 0.05 and |PearsonScore| > 0.05. In order to process the same simulated matrix as the other methods, we reimplemented these four steps in a Python method: (a) We first used a 250-kb window instead of 500 kb to be consistent with our chosen simulation setup and computed the Pearson correlation using the *correlation* function of the Python *NumPy* library. (b) In order to randomly select a set of enhancers E_r with similar accessibility to a given enhancer e_{ref} and its accessibility a_{ref}, we then computed the probability $P(e|e_{\text{ref}})$ of each $e \in E$ and their corresponding accessibility mean a_e to be in E_r using the normed kernel function and $sigma = 50$:

$$P(e|e_{\text{ref}}) = \frac{e^{-\text{sigma}.(a_e - a_{ref})^2}}{\sum_{e' \in E} e^{-\text{sigma}.(a_{e'} - a_{ref})^2}}$$

We randomly selected 200 enhancers using these probabilities and computed the Pearson score of each $e \in E_r$. All the peaks used in the simulation had the same length. (c) We transformed the Pearson correlations (of enhancers in the random set *Er* or the query enhancer e with the target gene expression) into a p-value using the cumulative distribution function *norm.cdf* from the *Scipy.stats* package, and excluded links with |Pearson| < 0.05 or p-value > 0.05.

SnapATAC

The SnapATAC methodology was described in the methods section of the SnapATAC paper [16]. It consisted of considering the expression of a gene g as a variable in a univariate logistic regression model that predicted the binary accessibility status of each enhancer within a 1-Mb window flanking g. For each enhancer e from the set of the flanking enhancers E, the method built a univariate logistic model $e = Logit(g)$ using the *glm* function with *link* = *'binomial'* from the *R* software and used a p-value cutoff

of 5e-8. We reimplemented the workflow of SnapATAC in a Python function using the *Logit* class with its *fit* method from the *statsmodels* Python library [52]. In the original study, a label-transfer procedure was conducted in order to identify the matching cells between two separate snATAC-seq and scRNA-seq datasets. However, this procedure was not necessary in our case because we derived the simulated matrices from a multi-omic snRNA-/snATAC-seq dataset for which we had a matching cell ID between the two modalities.

ArchR

The ArchR methodology to infer enhancer–promoter links from scATAC-seq and enhancer–gene correlations from single-cell multi-omic data was described in the supplementary methods of the original study [15]. The method first created a *low-overlapping aggregate of cells* which excludes any pair of cells within this aggregate that share more than 80% of accessible peaks. This was done by first computing a K-nearest neighbor clustering ($K = 100$ with the Euclidean distance) for a subset of 500 random reference cells (*NS*) and using a 2D embedding of the cells as input. The method iteratively removed cells having > 80% of similarities with regard to the KNN neighbors and aggregated the cell vectors of the remaining cells based on their KNN neighbors, with the values further scaled and log-transformed. The Pearson correlation was then computed between the target enhancer and promoter accessibilities, and a *p*-value was inferred by (a) transforming the correlation score into a t-statistic: $t\text{Stat} = \frac{|\text{score}|}{\sqrt{1-\max(\text{score}^2, \frac{1e^{-17}}{(K-2)})}}$ and using the cumulative distribution function of the Student's *t* law to convert *tStat* into a *p*-value. Finally, an FDR correction was applied using the Benjamini–Hochberg procedure. The R source code of the peak*addCoaccessibility* method is available here: https://rdrr.io/github/GreenleafLab/ArchR/src/R/IntegrativeAnalysis.R and the C++ source code of the iterative aggregation strategy is available here: https://github.com/GreenleafLab/ArchR/blob/master/src/KNN_Utils.cpp. In order to process the same matrices in the same conditions as the other methods, we reimplemented the ArchR strategy in a Python script using the following modifications: (a) we used a Harmony + UMAP embedding instead of the iterative LSI strategy used in the original ArchR workflow. The embedding process was the same as described for Cicero. (b) We reimplemented the aggregate strategy from the R and the C++ scripts and used the *NearestNeighbors* from the *Scikit-Learn* Python library class with the *cosine* distance to compute the K-neighbors and iteratively computed the Jaccard similarity to exclude cells from the aggregates. We used the *pairwise_distances* function from Scikit-learn to compute the Pearson scores and the *fdrcorrection* function from the *Statsmodels* library to compute the FDR.

Robustlink

The Robustlink methodology was described in the original study [17]. It requires single-cell RNA-seq data along with either single-cell ATAC-seq or single-cell methylation data. If the data are from the same sample but are not multi-omics, the first step is to identify similar cells with scFusion. Secondly, Robustlink infers meta-cell by first performing PCA dimension reduction followed by a KNN graph construction ($K = 30$) and then applies the Leiden algorithm to find cell communities. Each community

constitutes a meta-cell for which the pseudo-bulk of the cells within the community is used. The RNA-seq raw count of each community is normalized with log(CPM + 1) and the ATAC-seq raw count is normalized with log(TPM + 1). Then, the Robust link computes the Spearman correlation between each relevant enhancer and gene and assesses their significance by constructing two null distributions, shuffling either metacells or shuffling regions. We installed Robustlink using the instructions from the Git package and applied Robustlink on our artificial ATAC + RNA dataset with the following protocol. After loading the single-cell RNA and ATAC cell × features matrices, we transformed the RNA matrix on a new embedding with its 25 first components using the TruncatedSVD function from the Scikit-learn Python library. We didn't have to use scFusion since our data were multi-omics. We then computed a KNN graph with $K = 30$ and the Euclidean metric using the NearestNeighbors function from Scikit-Learn. We then partitioned the graph using the Leiden algorithm with the *RBConfigurationVertexPartition* as spartition type from the leidenalg Python package. We then create *metacells* from the ATAC and RNA matrices by aggregating the cells within each community in pseudo-bulk and by normalizing the raw read count. Finally, we called the *compute_enh_gene_corrs* and *get_significance_stats* from the robustlink Python package. WE used *dist_th = 1e6, bins = 501* and 0.20 as fdr cutoff. Furthermore, we tested an array of resolutions from 1.0 to 90.0. We also tested using all cells instead of inferring the *meta-cells*. In this case, each cell is a unique *meta-cell*. The robustlink workflow used is available as a Python script (see Software's availability).

Intersecting the inferred links with the PCHi-C data

The PCHi-C data are available on GEO (see "Availability of data and materials") with a limited access using GSE214107 as accession ID. We downloaded the *ibed* files, which are similar to *bedpe* files with the six first columns referring to two genomic locations, for each strain, merged them, and retained only the unique links.

Estimating accuracy with the PCHi-C data

We used the physical enhancer–promoter interactions from the PCHi-C datasets from the islet and adipose tissues as references to compute the overall accuracy scores (precision, recall, f1-score) of the links inferred with the Enlink, Cicero, and Chi2 + FDR workflows. For both the islet and adipose, we analyzed each cell population independently with each workflow. We then combined all the links obtained for a given dataset and workflow and estimated the overall precision, recall, and f1-score using the set of the PCHi-C links as true positives. We used the *enhtools* software with the *intersect3* option from the Enhlink software suite (see above) to find the intersecting links between the reference set and the results of the different workflows.

Enrichment analysis with EnhancerAtlas datasets

We obtained six reference datasets from EnhancerAtlas 2.0 [26]. comprising reference enhancers for the human heart, human left ventricle, mouse striatum, and mouse islet, as well as reference enhancer–gene interactions for the human left ventricle and mouse striatum. Subsequently, we converted the genomic coordinates from mm9 to mm10

and from hg19 to hg38 using the LiftOver tool (https://genome.ucsc.edu/cgi-bin/hgLiftOver) from the UCSC genome browser [53]. For each relevant cell type, we employed enhtools to tally the number of links inferred from that cell type by Enhlink, Cicero, or the Chi2 + FDR procedure, where the linked enhancer overlapped with at least one reference enhancer, or both the promoter and the linked enhancer overlapped with a reference enhancer-gene interaction. The enrichment score was then defined as the ratio between the number of intersecting links and the total number of links. We intersected the Heart dataset with the nine CARE cell populations, the left ventricle datasets with the vCM CARE population, the islet dataset with the ten islet cell populations, and the striatum datasets with the combined Drd1 + Drd2 populations.

Estimating batch effect with entropy measurements

We quantified the impact of technical batch effects by first reasoning that a true link should be distributed over sequencing batches (that are otherwise biologically similar). We used the information theory principle and computed the Shannon entropy of the link with regard to the sequencing library ID of the dataset. The library ID of the islet and adipose datasets were the ID of the 10 × Genomics sequencing runs and were regarded as variables associated with the batch effect. We first computed for each inferred link from an enhancer e to a gene g the link vector v_l such as

$$v_l = v(e \cap g) = v_e \circ v_g$$

We then computed the Shanon entropy of v_l with regard to the set of library IDs B:

$$E(v_l|B) = -\sum_{b \in B}\left(\frac{v_l.v_b}{\sum_{i \in v_l} i}\right).log\left(\sum_{b \in B}\left(\frac{v_l.v_b}{\sum_{i \in v_l} i}\right)\right)$$

Here, v_b corresponds to a binary vector indicating which cells belong to b. Finally, we plotted the batch effect entropy distribution using the *seaborn* Python library.

Inferring Enhlinks atlases for the islet and adipose tissues

For each tissue, we processed each cell type independently using the library ID, sex, diet, and genotype/strain as covariates. We used 0.01 as the p-value cutoff, with the *secondOrder* and the *uniformSampling* options to infer the covariate-specific linkages from a uniform distribution of the covariates within each bootstrap sample. For each bootstrap sample, we used a random subset of 66% of the features, a maximum of 4 explanatory features, a depth of 2, and a downsampling size of 15,000 cells. We created a binary cell × promoter sparse matrix by considering all the promoter regions of each gene and give the value 1 for a given cell if at least one read is found, 0 otherwise. This matrix is then used as M_{target} (see Additional file S1). We used the *ATACMatUtils* command with the *-use_symbol* option from the *ATACdemultiplex* package (https://gitlab.com/Grouumf/ATACdemultiplex) to create the matrix from the BED file containing the reads and barcode IDs. We then intersected the obtained linkage of each cell type with the PCHi-C links of either the adipose or the islet tissues using the *enhtools* software with the *-intersect3* option.

Inferring Enhlinks for the striatum tissue

We restricted the Enhlink analysis to the two neuron cell types expressing either *Drd1* or *Drd2*. We first performed two Enhlink analyses with the default parameters on the two neuron subtypes expressing either *Drd1* or *Drd2* using (a) the scATAC-seq matrix alone and (b) the scATAC-seq matrix for the enhancer features and the scRNA-seq matrix to infer enhancers linked to gene expression. We used the default parameters with the library ID as a covariate when using the scATAC-seq matrix only but extended the *max_features* and *depth* parameters to 6 and 4, respectively. We then intersected the ATAC and ATAC + RNA linkages of the *Drd1* and *Drd2* cell type using the *enhtools* executable with the *-intersect* option. In a second analysis aimed at identifying Drd1- or Drd2-specific linkages, we processed with Enhlink all the cells from either the *Drd1* or the *Drd2* neurons and used the library ID together with the cell type (Drd1 or Drd2) as covariates. We performed two processing steps using either the ATAC-seq data alone or the ATAC-seq data combined with the RNA-seq data that we further intersected with *enhtools*. We also processed the same ATAC-seqs dataset with Cicero using the protocol described above.

Preparing eQTL collections

Bulk RNA-sequencing for eQTL analysis was accessed and downloaded from the Churchill Lab QTL viewer (https://qtlviewer.jax.org/viewer/CheslerStriatum accessed 02/17/2023). This dataset represents striatum samples collected from individual mice ($n = 368$) from the Diversity Outbred genetic reference population (The Jackson Laboratory catalog #009376). The downloaded package includes a gene expression estimate matrix, genotype probabilities, kinship matrix, and metadata including sex. We performed eQTL mapping restricted to the set of genes differentially expressed between *Drd1* and *Drd2* neurons ($n = 96$ and 103, respectively). eQTL mapping was performed using a linear mixed model to account for kinship on normalized, transformed gene-level expression values using the "scan1" function in r/qtl2 [54], including sex as an additive covariate. Genes passing a filter requiring a local-eQTL with a LOD score greater than 8 ($n = 162$) were further included for SNP association using the "scan1snps" function in r/qtl2. All variants within a 1.5 LOD confidence interval were included for association analysis. Resulting variants with LOD score drop of 1.5 from the maximum were retained, along with their strain distribution pattern, to intersect with Enlink significant links and compare to haplotype effect pattern for the linked eQTL gene. The haplotype effects of the QTL for the candidate genes *Kcnb2*, *Gulp1*, and *Col25a1* were estimated using "scan1blup" from r/qtl2.

Intersecting Enhlinks from the striatum with eQTL

We formatted the results of the eQTL analysis described above as a BED file listing the genomic coordinates of the SNPs and intersected these regions with the enhancers for the Drd1 and Drd2 neuron subtypes that were correlated to both promoter accessibility and expression of their corresponding genes (see above). We then computed a *p*-value using the Mann–Whitney procedure to test if the overlapping enhancer presented differential accessibility between the genotypes. We used the *p*-values and the LOD score to identify the enhancers of *Kcnb2*, *Gulp1*, and *Col25a1*. For each enhancer

and their associated promoter and gene, we computed the barplot of accessibility (for the enhancer and promoter) or gene expression, using the four 10 × Genomics libraries as individual measurements.

Evaluating Enhlink and Cicero with the striatum dataset

We intersected the linkages obtained from Cicero and Enlink for the Drd1/Drd2 neuron cell types from the ATAC-seq data with the 96 and 103 marker genes of the Drd1 and Drd2 neurons, respectively. For each gene and its inferred enhancer, we computed a univariate logistic regression: $Ve = f(Vg)$, with Ve the boolean vector of the enhancer e and of size $1 \times cell$, indicating if e is accessible for each cell, and Vg the numerical vector indicating the scaled gene expression of g for each cell. We used the *Logit* class with its *fit* function from the Python *statsmodels* library to infer the p-value of each model. We then compared the $-log10(p\text{-}value)$ distribution of the enhancer from Enhlink and from Cicero for different cutoffs: 0.0, 0.1, 0.2, and 0.28. We chose 0.28 as a cutoff to obtain a number of links (704) similar to the number obtained with Enhlink (802). We used the Mann–Whitney test to assess the difference of the p-value distributions between the links from Enhlink and with Cicero results. Since no difference was observed between Enhlink and Cicero with 0.28 as cutoff, we applied a t-test in this case, assuming normality of the distributions.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13059-024-03374-9.

Additional file 1: Extended description of the Enhlink procedure.

Additional file 2: Combined supplement materials: Figure S1 Accuracy scores inferred from enhancer and promoter vectors. Figure S2 Detailed Enhlink analytical workflow. Figure S3 Proportion of positively and negatively correlated links. Figure S4 Processing time comparison. Figure S5 f1-score, precision, and recall computed from scATAC-seq data. Figure S6 Importance of hyperparameters on Enhlink accuracy. Figure S7 F1-score, precision and recall on simulated promoters and enhancers. Figure S8 Robustlink performances. Figure S9 Histogram distribution of the entropy score. Figure S10 Number of links per tissue (islet or adipose). Figure S11 Linkage enrichment scores. Figure S12 A Striatum UMAP projection and linkage score distributions. Figure S13 Comparison of enhancer-gene expression association. Figure S14 Chromatin accessibility of distal *Drd1*-associated enhancers. Table S1 Description of Enhlink's hyperparameters. Table S2 Description of the hyperparameter values used for the simulation experiment.

Additional file 3: Peer review history.

Acknowledgements

We gratefully acknowledge the contribution of the Single Cell Biology service, the Genome Technologies service, and cyberinfrastructure high-performance computing resources at The Jackson Laboratory for expert assistance in the work described herein. These shared services are supported in part by the JAX Cancer Center (P30 CA034196). We would like to also thank Dr. Mike Lloyd and Dr. Anuj Srivastava for their help in obtaining and processing the PCHi-C data, and Dr. Vivek M. Philip, Dr. Robyn Ball, and Leona Gagnon for managing and executing experimental designs.

Peer review information

Tim Sands was the primary editor of this article and managed its editorial process and peer review in collaboration with the rest of the editorial team.

Authors' contributions

OP and BSW envisioned this project. OP developed Enhlink and implemented the project. OP, WZ, DS, and CLB conducted the analyses. CLB performed the eQTL analysis. OP, BSW, and CLB wrote the manuscript with inputs from all authors. CNB developed mouse models and collected tissues used for PCHi-C studies. EJC supervised the eQTL collection analysis. CLB supervised the striatum tissue collection and experimental design. CS and AO performed the tissue collection. SD performed the single-cell multiome experiment.

Funding

Research reported in this publication was supported by The Jackson Laboratory Cube Initiative and the Jackson Laboratory *Scientific Support Internal Fund* (SSIF) mechanism with the project id: 19005–21-05. Further funding was provided, in part, by the National Institute of General Medical Sciences grant R35GM133724 to CLB and the Jackson Laboratory Director's Innovation Fund to CLB and EJC. P50 DA039841 supports the Center for Systems Neurogenetics of Addiction

to EJC. Data generation through the Single Cell Biology service was supported in part by the JAX Cancer Center (P30 CA034196). Additional support was also provided by The Jackson Laboratory Cube Initiative.

Availability of data and materials

The PCHi-C data used in this manuscript are currently available with limited access on request. The multi-omics striatum dataset is publicly accessible using the following accession number: GSE228530 [55]. The Linkage atlases for the islet [56], adipose [57], and striatum [58] datasets are available as Figshare datasets with the following DOIs: https://doi.org/https://doi.org/10.6084/m9.figshare.22336033.v1 (adipose) https://doi.org/https://doi.org/10.6084/m9.figshare.22335919.v1 (islet), and https://doi.org/https://doi.org/10.6084/m9.figshare.26461363.v1 (striatum).
Enhlink is written in Go and is freely available at https://gitlab.com/Grouumf/enhlinktools [59]. We also have compiled executables compatible with Linux x86_64 or OSX arm64 and available at: https://figshare.com/articles/software/Compiled_executables_for_Enhlink_/22807103 [60]. Python implementations of the Chi2, Chi2 + FDR, Signac, and ArchR procedure are available at https://gitlab.com/Grouumf/rna_atac_jax_int [61]. The custom robustlink workflow used is available as a Python executable here: https://gitlab.com/Grouumf/enhlinktools/-/blob/master/scripts/robustlink_analysis.py [62].

Declarations

Ethics approval and consent to participate

All animal experiments were approved by the animal care and use committee of The Jackson Laboratory (Animal Use Summary 16043).

Competing interests

The author(s) declare(s) that they have no competing interests.

Received: 11 May 2023 Accepted: 20 August 2024
Published online: 02 September 2024

References

1. Panigrahi A, O'Malley BW. Mechanisms of enhancer action: the known and the unknown. Genome Biol. 2021;22:1–30.
2. Heinz S, Romanoski CE, Benner C, Glass CK. The selection and function of cell type-specific enhancers. Nat Rev Mol Cell Biol. 2015;16:144–54.
3. Robson MI, Ringel AR, Mundlos S. Regulatory landscaping: how enhancer-promoter communication is sculpted in 3D. Mol Cell. 2019;74:1110–22.
4. Corradin O, Scacheri PC. Enhancer variants: evaluating functions in common disease. Genome Med. 2014;6:1–14.
5. Claringbould A, Zaugg JB. Enhancers in disease: molecular basis and emerging treatment strategies. Trends Mol Med. 2021;27:1060–73.
6. Lieberman-Aiden E, van Berkum NL, Williams L, Imakaev M, Ragoczy T, Telling A, et al. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. Science. 2009;326:289–93.
7. Schoenfelder S, Javierre B-M, Furlan-Magaril M, Wingett SW, Fraser P. Promoter capture Hi-C: high-resolution, genome-wide profiling of promoter interactions. J Vis Exp. 2018; Available from: https://doi.org/10.3791/57320.
8. Galitsyna AA, Gelfand MS. Single-cell Hi-C data analysis: safety in numbers. Brief Bioinform. 2021;22:bbab316.
9. Pliner HA, Packer JS, McFaline-Figueroa JL, Cusanovich DA, Daza RM, Aghamirzaie D, et al. Cicero predicts cis-regulatory DNA interactions from single-cell chromatin accessibility data. Mol Cell. 2018;71:858-71.e8.
10. Wang A, Chiou J, Poirion OB, Buchanan J, Valdez MJ, Verheyden JM, et al. Single-cell multiomic profiling of human lungs reveals cell-type-specific and age-dynamic control of SARS-CoV2 host genes. 2020 [cited 2022 Nov 21]; Available from: https://elifesciences.org/articles/62522.
11. Li YE, Preissl S, Hou X, Zhang Z, Zhang K, Qiu Y, et al. An atlas of gene regulatory elements in adult mouse cerebrum. Nature. 2021;598:129–36.
12. BRAIN Initiative Cell Census Network (BICCN). A multimodal cell census and atlas of the mammalian primary motor cortex. Nature. 2021;598:86–102.
13. Kamimoto K, Hoffmann CM, Morris SA. CellOracle: dissecting cell identity via network inference and in silico gene perturbation. bioRxiv. 2020 [cited 2022 Nov 21]. p. 2020.02.17.947416. Available from: https://www.biorxiv.org/content/https://doi.org/10.1101/2020.02.17.947416v3.abstract.
14. Stuart T, Srivastava A, Madad S, Lareau CA, Satija R. Single-cell chromatin state analysis with Signac. Nat Methods. 2021;18:1333–41.
15. Granja JM, Corces MR, Pierce SE, Bagdatli ST, Choudhry H, Chang HY, et al. ArchR is a scalable software package for integrative single-cell chromatin accessibility analysis. Nat Genet. 2021;53:403–11.
16. Fang R, Preissl S, Li Y, Hou X, Lucero J, Wang X, et al. Comprehensive analysis of single cell ATAC-seq data with SnapATAC. Nat Commun. 2021;12:1–15.
17. Xie F, Armand EJ, Yao Z, Liu H, Bartlett A, Margarita Behrens M, et al. Robust enhancer-gene regulation identified by single-cell transcriptomes and epigenomes. Cell Genomics. 2023 [cited 2024 Mar 12];3. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10363915/.
18. Poirion O, Baker CN, Kumar P, Daigle S, Bhattacharyya T, Schott W, Harder J, Seignon M, Gaca M, Braun M, Churchill GA, Flynn B, White B, Robson P, George J, Ansarullah, Skelly DA (2024) Multi-tissue single cell profiling of diabetes susceptibility and resilience models reveals divergent, genetically encoded responses to an obesogenic diet, *In Preparation*.

19. Lobo MK, Nestler EJ. The striatal balancing act in drug addiction: distinct roles of direct and indirect pathway medium spiny neurons. Front Neuroanat. 2011;5:41.
20. Yager LM, Garcia AF, Wunsch AM, Ferguson SM. The ins and outs of the striatum: role in drug addiction. Neuroscience. 2015;301:529–41.
21. Hocker JD, Poirion OB, Zhu F, Buchanan J, Zhang K, Chiou J, et al. Cardiac cell type-specific gene regulatory programs and disease risk association. Sci Adv. 2021;7. Available from: https://doi.org/10.1126/sciadv.abf1444.
22. McArthur E, Capra JA. Topologically associating domain boundaries that are stable across diverse cell types are evolutionarily constrained and enriched for heritability. Am J Hum Genet. 2021;108:269–83.
23. Vandereyken K, Sifrim A, Thienpont B, Voet T. Methods and applications for single-cell and spatial multi-omics. Nat Rev Genet. 2023;1–22.
24. Dymora P, Paszkiewicz A. Performance analysis of selected programming languages in the context of supporting decision-making processes for industry 4.0. NATO Adv Sci Inst Ser E Appl Sci. 2020;10:8521.
25. Website. Available from: https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.
26. Gao T, Qian J. EnhancerAtlas 2.0: an updated resource with enhancer annotation in 586 tissue/cell types across nine species. Nucleic Acids Res. 2019;48:D58-64.
27. Saul MC, Philip VM, Reinholdt LG. Center for Systems Neurogenetics of Addiction, Chesler EJ. High-diversity mouse populations for complex traits. Trends Genet. 2019;35:501–14.
28. Gerfen CR, Engber TM, Mahan LC, Susel Z, Chase TN, Monsma FJ Jr, et al. D1 and D2 dopamine receptor-regulated gene expression of striatonigral and striatopallidal neurons. Science. 1990;250:1429–32.
29. Churchill GA, Gatti DM, Munger SC, Svenson KL. The diversity outbred mouse population. Mamm Genome. 2012;23:713.
30. Website. Available from: https://www.cell.com/trends/genetics/fulltext/S0168-9525(19)30065-4.
31. Lu KM, Evans SM, Hirano S, Liu FC. Dual role for Islet-1 in promoting striatonigral and repressing striatopallidal genetic programs to specify striatonigral cell identity. Proc Natl Acad Sci U S A. 2014 [cited 2022 Nov 30];111. Available from: https://pubmed.ncbi.nlm.nih.gov/24351932/.
32. Yang F, Wang J, The GTEx Consortium, Pierce BL, Chen LS. Identifying cis-mediators for trans-eQTLs across many human tissues using genomic mediation analysis. Genome Res. 2017;27:1859.
33. Liu H, Zhou J, Tian W, Luo C, Bartlett A, Aldridge A, et al. DNA methylation atlas of the mouse brain at single-cell resolution. Nature. 2021;598:120–8.
34. Grosselin K, Durand A, Marsolier J, Poitou A, Marangoni E, Nemati F, et al. High-throughput single-cell ChIP-seq identifies heterogeneity of chromatin states in breast cancer. Nat Genet. 2019;51:1060–6.
35. Dimitriu MA, Lazar-Contes I, Roszkowski M, Mansuy IM. Single-cell multiomics techniques: from conception to applications. Front Cell Dev Biol. 2022;10: 854317.
36. Breiman L. Random Forests. Mach Learn. 2001;45:5–32.
37. Zhang K, Hocker JD, Miller M, Hou X, Chiou J, Poirion OB, et al. A single-cell atlas of chromatin accessibility in the human genome. Cell. 2021;184:5985-6001.e19.
38. Umans BD, Battle A, Gilad Y. Where are the disease-associated eQTLs? Trends Genet. 2021;37:109.
39. Gamazon ER, Wheeler HE, Shah KP, Mozaffari SV, Aquino-Michaels K, Carroll RJ, et al. A gene-based association method for mapping traits using reference transcriptome data. Nat Genet. 2015;47:1091–8.
40. Poirion O, Zhu X, Ching T, Garmire LX. Using single nucleotide variations in single-cell RNA-seq to identify subpopulations and genotype-phenotype linkage. Nat Commun. 2018;9:1–13.
41. Website. Available from: https://doi.org/10.1002/ame2.12032.
42. Chen P-T, Zoller B, Levo M, Gregor T. Gene activity as the predictive indicator for transcriptional bursting dynamics. ArXiv. 2023; Available from: https://www.ncbi.nlm.nih.gov/pubmed/37131882.
43. Horvathova I, Voigt F, Kotrys AV, Zhan Y, Artus-Revel CG, Eglinger J, et al. The dynamics of mRNA turnover revealed by single-molecule imaging in single cells. Mol Cell. 2017;68:615-25.e9.
44. Kang HM, Subramaniam M, Targ S, Nguyen M, Maliskova L, McCarthy E, et al. Author Correction: Multiplexed droplet single-cell RNA-sequencing using natural genetic variation. Nat Biotechnol. 2020;38:1356.
45. Hao Y, Hao S, Andersen-Nissen E, Mauck WM 3rd, Zheng S, Butler A, et al. Integrated analysis of multimodal single-cell data. Cell. 2021;184:3573-87.e29.
46. McGinnis CS, Murrow LM, Gartner ZJ. DoubletFinder: doublet detection in single-cell RNA sequencing data using artificial nearest neighbors. Cell Syst. 2019;8:329-37.e4.
47. Korsunsky I, Millard N, Fan J, Slowikowski K, Zhang F, Wei K, et al. Fast, sensitive and accurate integration of single-cell data with Harmony. Nat Methods. 2019;16:1289–96.
48. Saunders A, Macosko EZ, Wysoker A, Goldman M, Krienen FM, de Rivera H, et al. Molecular diversity and specializations among the cells of the adult mouse brain. Cell. 2018;174:1015-30.e16.
49. Zhang Y, Liu T, Meyer CA, Eeckhoute J, Johnson DS, Bernstein BE, et al. Model-based analysis of ChIP-Seq (MACS). Genome Biol. 2008;9:1–9.
50. Quinlan JR. Induction of decision trees. Mach Learn. 1986;1:81–106.
51. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: machine learning in Python. J Mach Learn Res. 2011;12:2825–30.
52. Seabold S, Perktold J. Statsmodels: Econometric and statistical modeling with python. Proceedings of the 9th Python in Science Conference. SciPy; 2010. Available from: https://conference.scipy.org/proceedings/scipy2010/seabold.html.
53. Raney BJ, Barber GP, Benet-Pagès A, Casper J, Clawson H, Cline MS, et al. The UCSC Genome Browser database: 2024 update. Nucleic Acids Res. 2024;52:D1082–8.
54. Broman KW, Gatti DM, Simecek P, Furlotte NA, Prins P, Sen Ś, Yandell BS, Churchill GA. R/qtl2: Software for Mapping Quantitative Trait Loci with High-Dimensional Data and Multiparent Populations. Genetics. 2019;211(2):495–502.
55. Zuo W, Spruce C, Poirion O, White B, Baker C. Single-cell co-profiling of gene expression and chromatin accessibility in the mice striatum with varied genetic backgrounds. Gene Expression Omnibus. GSE228530. 2023.

56. Poirion O, Spruce C, Chesler EJ, Zuo W, Daigle S, Wilson A, et al. Enhancer-promoter linkage atlas for the mouse Islet tissue generated with enhlink. Insights into type II diabetes mechanisms and covariate-specific regulation. figshare; 2023. Available from: https://figshare.com/articles/dataset/Enhancer-Promoter_Linkage_Atlas_for_the_Mouse_Islet_tissue_generated_with_Enhlink_Insights_into_Type_II_Diabetes_Mechanisms_and_Covariate-Specific_Regulation/22335919/1.
57. Poirion O, Zuo W, Spruce C, Daigle S, Wilson-Smith A, Baker C, et al. Enhancer-promoter linkage atlas for the mouse adipose tissue generated with enhlink. Insights into type II diabetes mechanisms and covariate-specific regulation. figshare; 2023. Available from: https://figshare.com/articles/dataset/Enhancer-Promoter_Linkage_Atlas_for_the_Mouse_Adipose_tissue_generated_with_Enhlink_Insights_into_Type_II_Diabetes_Mechanisms_and_Covariate-Specific_Regulation/22336033/1.
58. Poirion O, Zuo W, Spruce C, Daigle S, Wilson-Smith A, Skelly DA, et al. Multi-omics linkage analysis for the striatopallidal (*Drd1*) and the striatopallidal (*Drd2*) neurons from huamn striatum. figshare; 2024. Available from: https://figshare.com/articles/dataset/Multi-omics_linkage_analysis_for_the_striatopallidal_i_Drd1_i_and_the_striatopallidal_i_Drd2_i_neurons_from_huamn_striatum/26461363/1.
59. Poirion O. EnhLinkTools. GitLab. https://gitlab.com/Grouumf/enhlinktools. 2023.
60. Poirion O. Compiled executables for Enhlink. figshare.https://figshare.com/articles/software/Compiled_executables_for_Enhlink_/22807103. 2023.
61. Poirion O. GitLab. https://gitlab.com/Grouumf/rna_atac_jax_int. 2023.
62. Poirion O. robustlink_analysis.py. GitLab. https://gitlab.com/Grouumf/enhlinktools/-/blob/master/scripts/robustlink_analysis.py. 2023.

Publisher's Note