

École polytechnique de Louvain

Benchmarking Machine Learning Architectures for Gene Regulatory Network Inference

Author: **Emile VAN GYSEL**
Supervisor: **Estelle MASSART**
Readers: **Florence BOSCO, Sébastien JODOGNE**
Academic year 2025–2026
Master [120] in Computer Science

Abstract

Gene regulatory networks (GRNs) capture the regulatory interactions that govern gene expression, providing insight into how cellular identity is established and how dysregulation of these interactions can drive disease. This makes GRNs a valuable tool in drug discovery, where identifying the regulators responsible for aberrant gene expression can help prioritise therapeutic targets.

Inferring these networks is difficult, as no complete experimentally validated reference of true regulatory interactions exists, making evaluation of GRN inference methods challenging. GRETA, a recently developed benchmarking framework, addresses this by comparing inferred networks through similarity and stability, and evaluating them against three complementary families of metrics capturing causal, predictive, and literature-based evidence.

This thesis investigates whether deep learning can improve GRN inference by integrating two neural network-based methods, LINGER and DeepMAPS, into GRETA, and benchmarking them against existing inference methods on peripheral blood mononuclear cell data.

The results suggest that deep learning does not systematically outperform existing methods. LINGER remains competitive across most evaluation tasks and its transcription factor activity module shows substantial agreement with SCENIC+, the current state-of-the-art method. DeepMAPS underperforms across most evaluation tasks, which may partly reflect the difficulties encountered in reproducing its original implementation.

Acknowledgements

First and foremost, I would like to thank my supervisor Estelle Massart for agreeing to supervise this thesis and for opening the door to AQYLA, the biotechnology company where this work was carried out. That opportunity allowed me to discover the biotech world and to meet many remarkable people along the way.

I am particularly grateful to Florence Bosco for her guidance throughout this work, for the valuable feedback she provided at every stage, and for introducing me to many of her colleagues in the team, as well as for agreeing to read this thesis.

I am equally grateful to Gabriele Partel for the stimulating discussions around the aims of this work, his help on the GRN inference aspects, and for reading the final manuscript. I would also like to thank Sébastien Jodogne for agreeing to read this thesis.

I would also like to thank Vincent Wullaume and Kacper Olszewski for their careful reading and constructive feedback.

Finally, I owe a great deal to my family. Their unconditional support, patience, and encouragement throughout these years of study have been a constant source of motivation, and this thesis would not have been possible without them.

Computational resources have been provided by the Consortium des Équipements de Calcul Intensif (CÉCI), funded by the Fonds de la Recherche Scientifique de Belgique (F.R.S.-FNRS) under Grant No. 2.5020.11 and by the Walloon Region.

Use of AI

Artificial intelligence tools were used in four ways during the preparation of this thesis. First, as a search aid to navigate code documentation and locate relevant research papers. Second, as a language assistant to rephrase, improve clarity, or synthesise written content. Third, to assist in generating figures and diagrams, either by producing Python code for data plots or TikZ code for schematic diagrams. Fourth, during the integration of the methods into GRETA, AI tools were used to suggest implementation ideas and help debug methods that would not run, such as which libraries or tools to consider for a given task and how to use them effectively; the resulting code was then reviewed, adapted, and validated by the author.

Whenever a figure was produced with AI assistance, it is marked with the symbol ✱ in its caption.

Contents

1	From gene expression to GRNs	1
1.1	Gene expression	1
1.2	Gene Regulatory Networks	2
1.3	The input data	3
1.4	GRNs as a Tool for Drug Discovery	5
2	State of the Art	6
2.1	From Bulk to Single-Cell Multi-Omics	6
2.1.1	Bulk transcriptomics	6
2.1.2	Bulk multi-omics	7
2.1.3	Single-cell multi-omics	7
2.2	Classes of GRN inference methods	7
2.2.1	Correlation-Based Methods	7
2.2.2	Regression-Based Methods	8
2.2.3	Dynamical Systems Methods	9
2.2.4	Deep Learning Methods	9
3	LINGER	11
3.1	Introduction	11
3.2	Priors	11
3.3	Preprocessing	11
3.4	Training	12
3.5	Outputs	13
3.6	Baseline version	14
3.7	Transcription Factor activity	14
3.8	Parallelisation	15
3.8.1	Preprocessing and training	15
3.8.2	GRN reconstruction	16
4	DeepMAPS	17
4.1	Introduction	17
4.2	Preprocessing	17
4.3	Training	18
4.3.1	Initial embeddings	18
4.3.2	Heterogeneous graph transformer model	18
4.3.3	Embedding update process	18
4.3.4	Attention scores	20

4.3.5	Loss	20
4.4	Outputs	20
4.4.1	Cell clustering	20
4.4.2	Functional gene modules	21
4.4.3	From gene modules to regulons	21
4.4.4	Cell-type-specific GRNs	22
4.5	Issues Encountered with the functional gene modules	23
5	GRETA	25
5.1	Introduction	25
5.2	Adding a new dataset	26
5.3	Adding new methods	27
5.3.1	Decoupling	27
5.3.2	LINGER integration	29
5.3.3	DeepMAPS integration	29
5.3.4	Random baseline	31
5.4	Evaluation metrics	32
5.4.1	Mechanistic	32
5.4.2	Predictive	34
5.4.3	Literature based	36
6	Benchmark	37
6.1	LINGER triplets extraction strategy	37
6.2	LINGER baseline	39
6.3	Similarity	40
6.4	Stability	42
6.4.1	Reproducibility across runs	42
6.4.2	Robustness to data downsampling	42
6.5	GRN evaluation on the PBMC dataset	44
6.5.1	Choice of the dataset	44
6.5.2	Global performance overview	44
6.5.3	Best methods per category	46
6.6	TF activity	48
7	Conclusion	51
7.1	Summary	51
7.2	Limitations	51
A	Appendix	53
A.1	Categories of GRN inference methods	53
A.2	PBMC dataset	54
A.3	GRETA Evaluation metrics and processed databases	55
A.4	GRETA Definition of evaluation metrics	56
A.5	Stability of GRETA methods	57
A.6	TF Activity Correlation Between LINGER and SCENIC+	58

List of Figures

1.1	Overview of gene expression, showing transcription and translation processes. From [1].	1
1.2	Overview of gene transcription. From [2].	2
1.3	Multi-omics data. From [2].	3
2.1	Parallel development of sequencing technologies and GRN inference methods. From [3].	6
2.2	Schematic of FigR. From [3].	8
2.3	Schematic of SCENIC+. From [3].	9
2.4	Schematic of Dictys. From [3].	9
2.5	Schematic of DeepMAPS and LINGER. From [3].	10
3.1	LINGER per-gene neural network architecture ✱	13
3.2	Benchmark of LINGER parallelisation on the PBMC dataset ✱.	16
4.1	Single HGT layer: per-node embedding update via multi-head attention. From [4].	19
5.1	Graphical abstract of GRETA. From [5]	25
5.2	DAG for dataset integration	26
5.3	AnnData schema representation. From [6].	27
5.4	GRN inference steps. From [5]	28
5.5	DAG for LINGER integration	29
5.6	Perturbation propagation workflow ✱.	34
5.7	TF regulon and matched gene sets ✱.	35
5.8	Literature-based (grey) and genomic (green) validation strategies for GRN evaluation ✱.	36
6.1	Evaluation of the triplet-based GRN construction strategy from LINGER outputs ✱	38
6.2	Baseline comparison of LINGER and its baseline across Predictive and Genomic tasks ✱.	39
6.3	Network size statistics across GRN inference methods	40
6.4	Pairwise overlap coefficients between GRNs at the TF, CRE, and target gene levels	41
6.5	Pairwise overlap coefficients between GRNs at the edge level	41
6.6	Pairwise overlap coefficients between five LINGER runs on the full pituitary gland dataset at the TF, CRE, gene, and edge levels ✱.	42

6.7	Overlap coefficients between GRNs inferred from downsampled pituitary gland dataset and the one inferred using all available data *	43
6.8	Stability scores (AUC) of LINGER *	43
6.9	$F_{0.1}$ scores across 10 evaluation tasks grouped by metric category for 12 GRN inference methods *	45
6.10	Number of gold, silver, and bronze medals obtained by each method across all evaluation tasks *	46
6.11	Mean $F_{0.1}$ scores per GRN inference method, grouped by metric category *	47
6.12	Heatmaps of TF activity z-scores for the 54 TFs shared between LINGER and SCENIC+ *	50
A.1	Summary of current GRN inference methods for paired multi-omic data included in this paper *	53
A.2	PBMC dataset cell type distribution *	54
A.3	Evaluation metrics and processed databases. From [5].	55
A.4	Definition of evaluation metrics. From [5].	56
A.5	Overlap coefficients of TFs (left) and target genes (right) between three separate runs per method.	57
A.6	Overlap coefficients and stability scores for GRETA methods.	57
A.7	Spearman correlation of TF activity z-scores between LINGER and SCENIC+ *	58

List of Abbreviations

AE	Autoencoder
ATAC-seq	Assay for Transposase-Accessible Chromatin Sequencing
AUC	Area Under the Curve
CRE	Cis-Regulatory Element
CRP	Cis-Regulatory Potential
DAG	Directed Acyclic Graph
DeepMAPS	Deep learning-based Multi-omics Analysis Platform for Single-cell data
DNA	Deoxyribonucleic Acid
DORC	Domain Of Regulatory Chromatin
EWC	Elastic Weight Consolidation
FDR	False discovery rate
GAS	Gene Activity Score
GEO	Gene Expression Omnibus
GRETA	Gene Regulatory nETwork Analysis
GRN	Gene Regulatory Network
HGT	Heterogeneous Graph Transformer
KL	Kullback–Leibler (divergence)
KNN	k -Nearest Neighbour
LINGER	Lifelong Neural network for GEne Regulation
Mb	Megabase
MLP	Multi-Layer Perceptron
mRNA	Messenger RNA
PBMC	Peripheral Blood Mononuclear Cell
PCA	Principal Component Analysis
RNA	Ribonucleic Acid
RNA-seq	RNA Sequencing
scATAC-seq	Single-Cell Assay for Transposase-Accessible Chromatin Sequencing
scRNA-seq	Single-Cell RNA Sequencing
SCENIC+	Single Cell rEgulatory Network Inference and Clustering
SHAP	SHapley values
SFP	Steiner Forest Problem
SNN	Shared Nearest Neighbour
TF	Transcription Factor
TFB	Transcription Factor Binding score
TG	Target Gene
tRNA	Transfer RNA
TRP	Trans-Regulatory Potential
TSS	Transcription Start Site
ULM	Univariate Linear Model
WNN	Weighted Nearest Neighbours

1. From gene expression to GRNs

1.1 Gene expression

Genes are segments of DNA that contain the information to make proteins: chains of amino acids required for the structure, function and regulation of the body's tissues and organs. Each cell contains all the DNA of an organism, known as its genome, organised in about 20,000 different genes [7].

Proteins are produced through gene expression (Figure 1.1). The first step is transcription, in which the DNA sequence of a gene is copied into a single-stranded RNA molecule called messenger RNA (mRNA) [8]. Unlike DNA, mRNA can be read by the cell's machinery and can carry genetic information out of the nucleus¹. The mRNA then moves to the ribosomes, the site of protein synthesis, where translation takes place.

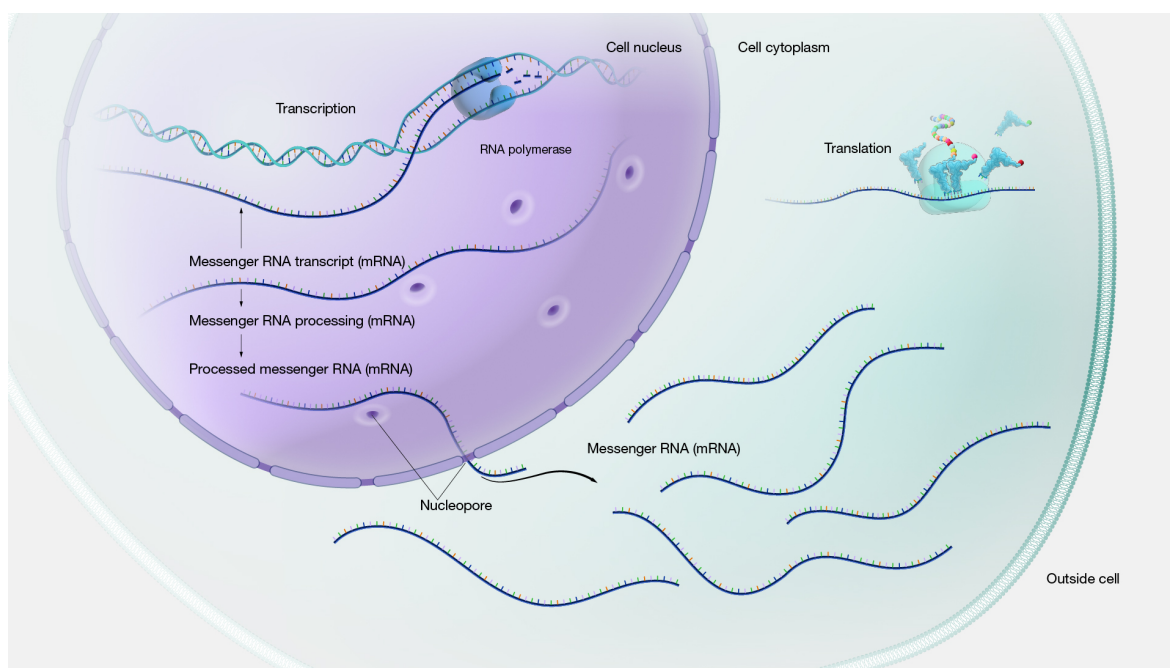


Figure 1.1: Overview of gene expression, showing transcription and translation processes. From [1].

¹membrane-enclosed organelle within a cell that contains the chromosomes [9]

An mRNA strand is built from four nucleotides: adenine (A), cytosine (C), guanine (G), and uracil (U). The sequence is read in groups of three bases known as codons, and each codon encodes a particular amino acid. During translation, each codon in the mRNA pairs with the complementary anticodon on a transfer RNA (tRNA) molecule, which delivers the corresponding amino acid. As this process repeats along the strand, the amino acids are linked together into the growing chain that will fold into the finished protein [10].

1.2 Gene Regulatory Networks

For a gene to be expressed, it first has to be transcribed (Figure 1.2). Understanding what regulates gene transcription is the essence of gene regulatory networks. Transcription is carried out by RNA polymerase, the enzyme that synthesises mRNA from a DNA template, and its activity is controlled through the interaction between cis- and trans-regulatory elements.

Cis-regulatory elements (CREs) are DNA sequences located on the same DNA molecule as the gene they regulate, where regulatory proteins can bind. They come in three main types: promoters² located directly upstream of the gene, as well as enhancers and silencers³ located further away.

Trans-regulatory elements, by contrast, are diffusible molecules, most prominently transcription factors (TFs), which act on one or more target genes (TGs) by binding to their corresponding cis-regulatory elements [13].

Together with cofactors that assist in assembling the transcriptional machinery, transcription factors regulate gene expression by influencing the recruitment and activity of RNA polymerase [2]. In this way, they can promote transcriptional initiation, acting as activators, or reduce transcriptional activity, acting as repressors. The combined action of TFs and cofactors therefore determines whether a gene is ultimately transcribed and expressed.

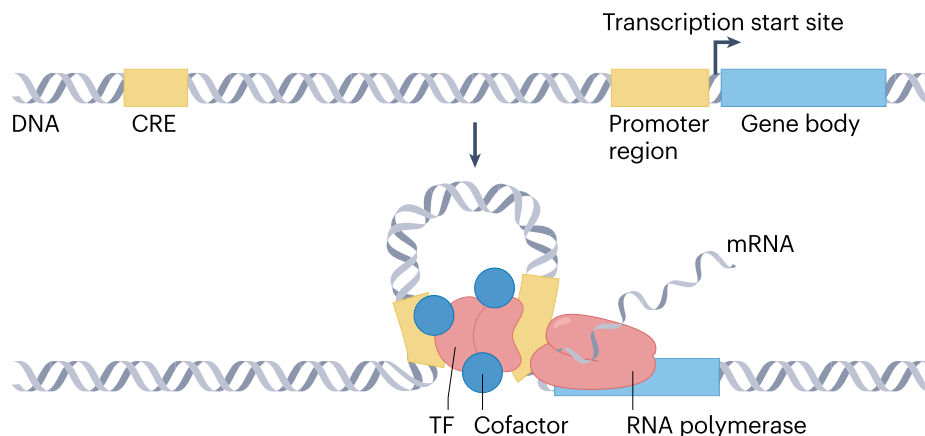


Figure 1.2: Overview of gene transcription. From [2].

²A regulatory region in the genome located before the transcriptional start site of a gene, where RNA polymerase binds to initiate transcription [11].

³Distal regulatory DNA regions where transcription regulatory proteins can bind to respectively activate or repress transcription [12].

Gene regulatory networks can be represented as graphs in which nodes correspond to transcription factors (TFs), regulatory elements (CREs), or target genes (TGs), and edges represent three types of interactions:

1. TF–CRE edges, the TF binding strength α
2. CRE–TG edges, the cis-regulatory strength β
3. TF–TG edges, the trans-regulatory strength γ

To understand how a particular target gene is regulated, it is necessary to identify both the cis-regulatory elements associated with it and the transcription factors that bind to these regions. CRE–TG interactions capture the link between regulatory elements and their target genes, often inferred from genomic proximity, while TF–CRE interactions describe the binding of TFs to accessible regulatory regions. Together, these relationships determine how TF activity influences gene expression, which is represented by TF–TG interactions. These three interaction types therefore capture complementary layers of regulation and together form the basis of a gene regulatory network.

Since transcription factors are proteins encoded by genes and produced through gene expression, all transcription factors are also target genes, but not all target genes are transcription factors. This implies that TF–TF interactions can also occur within gene regulatory networks.

1.3 The input data

Gene regulatory networks are inferred from single-cell multi-omics data (Figure 1.3), which combine two paired layers of molecular information measured in the same cells.

The first layer is the transcriptome, which captures gene expression using single-cell RNA sequencing (scRNA-seq). This technology measures the abundance of RNA molecules present in a cell at a given time, indicating which genes are currently being transcribed [14]. Although every cell contains the same set of genes, different cell types show different patterns of gene expression [15].

In practice, scRNA-seq data is represented as a gene expression matrix, where columns correspond to genes and rows correspond to individual cells. Each entry in the matrix contains raw counts, representing the number of RNA molecules (or sequencing reads) associated with a given gene detected in a specific cell. Each cell is identified by a unique DNA sequence called a cell barcode, allowing all reads from the same cell to be grouped together into one row of the matrix.

Cell type annotation consists of mapping each barcode to a cell type, and since both modalities share the same barcodes, the same annotation is used for both. This

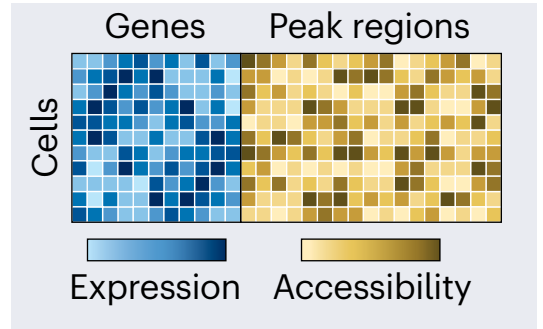


Figure 1.3: Multi-omics data. From [2]

annotation is provided as input to the GRN inference methods, supplying additional information by grouping individual single cells into broader cell-type categories while still preserving single-cell resolution.

The second layer captures chromatin accessibility, measured by the single-cell Assay for Transposase-Accessible Chromatin sequencing (scATAC-seq). Chromatin is essentially DNA wrapped around histone proteins, and its accessibility depends on how tightly it is packaged: tightly packed regions are hidden from the transcriptional machinery, whereas loosely packed regions are exposed [16].

scATAC-seq identifies these open regions by sequencing short DNA fragments extracted from accessible chromatin, revealing which regions are accessible. These open regions typically correspond to active cis-regulatory elements where transcription factors can bind.

In practice, scATAC-seq data is represented as a chromatin accessibility matrix, where columns correspond to accessible genomic regions (called peaks) and rows correspond to individual cells.

These peaks are identified by first pooling the sequencing fragments from all cells, then running a peak-calling algorithm such as MACS3 [17], which scans the genome and flags regions where fragments pile up significantly more than expected by chance, indicating that those regions are consistently accessible across cells. Each entry in the matrix then contains raw counts corresponding to the number of sequencing fragments overlapping a given peak in a specific cell. As a result, peaks with higher accessibility tend to accumulate more fragments across cells, reflecting their regulatory activity in a given cellular context.

Both matrices are then processed differently depending on the gene regulatory network inference method. However, common preprocessing steps are applied to both.

Low-quality features and cells are first removed: genes or peaks detected in very few cells are discarded, and cells expressing only a small number of features are removed as well. The remaining count matrices are then normalised to account for differences in sequencing depth across cells. In scRNA-seq, sequencing depth refers to the total number of RNA molecules captured per cell, while in scATAC-seq it corresponds to the total number of sequencing fragments per cell.

After normalisation, both matrices are log-transformed to stabilise variance and reduce the influence of highly expressed genes or highly accessible peaks. To reduce dimensionality, only the most informative features are retained: the top highly variable genes for RNA and the top highly variable regions for ATAC. Finally, only cells present in both modalities are kept, ensuring that each cell has paired RNA and ATAC measurements.

In summary, scRNA-seq quantifies the expression of both TFs and their target genes (TGs), while scATAC-seq reveals the active regulatory regions where regulation takes place. Combining the two provides the information needed to reconstruct TF-CRE-TG triplets and infer GRNs, with the assumption that true regulatory interactions can be observed in molecular data [2].

1.4 GRNs as a Tool for Drug Discovery

GRNs are highly useful in drug discovery, where the goal is to identify drug targets: proteins (such as TFs) whose activity can be modulated to restore a healthy gene expression state in diseased cells. Many diseases are fundamentally disorders of gene regulation: genetic⁴ and epigenetic⁵ alterations dysregulate transcription factors and their downstream targets, producing aberrant protein activity that drives pathological cell states.

Cancer is a telling example: Gonda and Ramsay argue that the dysregulation of key transcriptional regulators not only defines the cancer phenotype⁶ but is essential for its development and maintenance [21], making TFs particularly compelling therapeutic candidates.

In practice, identifying such targets is extremely challenging: only 10% of drug candidates entering clinical trials will reach the market, primarily due to lack of efficacy and toxicity [22], reflecting the complexity of diseases and the difficulty of identifying the right target [23]. Systematic characterisation of the regulatory landscape through GRNs can help address this: by revealing which TFs and CREs control a disease-relevant gene, GRNs provide mechanistic evidence that a candidate target is genuinely embedded in the pathological regulatory programme, rather than merely co-expressed with disease markers.

The topology of a GRN can be exploited in several complementary ways for drug discovery. First, hub identification highlights TFs that act as master regulators: highly connected nodes that control the expression of large sets of target genes. Such master regulators have proven to be valuable therapeutic targets: the core regulatory circuitry of lineage-specific TFs⁷ is hijacked during oncogenesis⁸, resulting in cancer cell fate being maintained by these same factors [24], making them attractive candidates for therapeutic disruption.

Second, differential network analysis compares GRNs across conditions, between healthy and diseased states, or before and after treatment, and exposes the regulatory rewiring events that distinguish one biological state from another. This is particularly relevant for understanding treatment resistance, where regulatory programmes reorganise to bypass targeted therapies [23].

Finally, GRNs enable in silico perturbation: by changing the expression of a candidate TF and propagating the effect through the network, one can simulate how its target genes would respond [2, 25], allowing researchers to rank and prioritise candidate targets before committing to experimental validation.

⁴Genetic alterations refer to permanent changes in the DNA sequence itself, such as mutations [18].

⁵Epigenetic alterations are changes in gene expression that occur without modification of the underlying DNA sequence, driven by environmental factors or cellular context [19].

⁶Observable characteristics of cancer cells, including uncontrolled proliferation or invasion of surrounding tissues [20].

⁷TFs whose expression is restricted to a particular cell type.

⁸Process by which normal cells transform into cancer cells through the accumulation of genetic and epigenetic alterations.

2. State of the Art

2.1 From Bulk to Single-Cell Multi-Omics

Chapter 1 introduced GRNs as networks inferred from single-cell multi-omics data, but this has not always been the case. GRN inference has evolved hand in hand with sequencing technologies, and the single-cell matched paradigm is only the most recent stage of a longer progression. Figure 2.1 illustrates this parallel development.

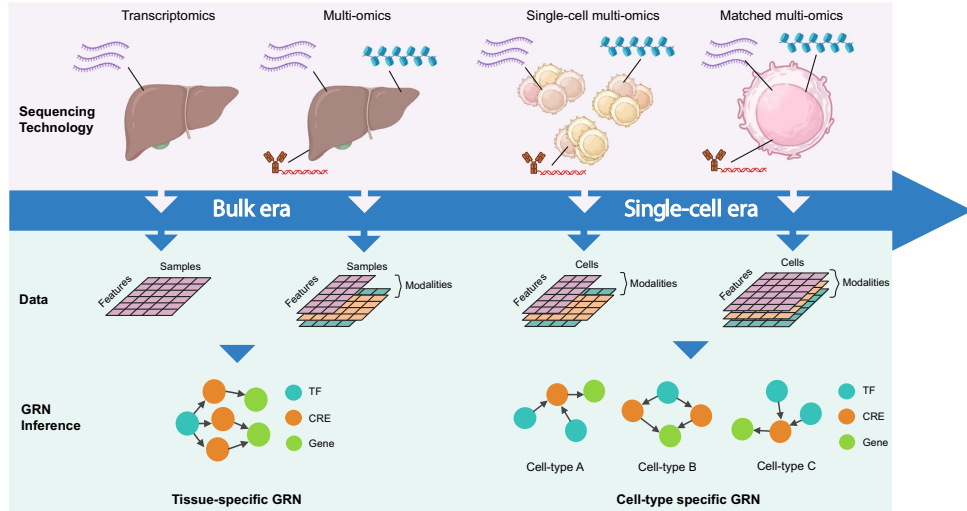


Figure 2.1: Parallel development of sequencing technologies and GRN inference methods. From [3].

2.1.1 Bulk transcriptomics

The earliest GRN inference methods operated on bulk RNA-seq data (Figure 2.1, first column), where gene expression is averaged across millions of cells from a tissue sample. The central idea was guilt by association: transcription factors and target genes that are co-expressed across samples are assumed to be functionally linked. While this produced the first large-scale regulatory networks, the approach suffered from two fundamental limitations. First, averaging over a heterogeneous cell population erases all cell-type-specific regulatory information. Second, transcriptomics data captures only RNA abundance, missing the chromatin accessibility layer that reveals whether transcription can occur at all.

2.1.2 Bulk multi-omics

Integrating chromatin accessibility data (ATAC-seq) alongside RNA-seq (Figure 2.1, second column) addressed the second limitation. Regulatory interactions could now be anchored to accessible genomic regions, introducing a mechanistic constraint: a TF must bind to an open CRE to regulate its target gene. This enabled the inference of TF–CRE–TG triplets rather than simple co-expression links. However, since measurements still aggregate all cells in the tissue, the core problem of cellular heterogeneity remained unsolved.

2.1.3 Single-cell multi-omics

Single-cell sequencing technologies (Figure 2.1, two last columns) resolved the heterogeneity problem by profiling gene expression and chromatin accessibility at single-cell resolution, enabling more accurate and comprehensive GRN inference [3]. Profiling both modalities within the same cell, as enabled by technologies such as 10x Multiome [26], ensures they share the same cell barcodes, avoiding the need for cross-modal alignment¹. The methods reviewed in this chapter span both multimodal approaches² that take such paired data as input, and unimodal approaches³ that operate on a single modality.

2.2 Classes of GRN inference methods

GRN inference methods can be divided into four main classes (Appendix A.1). Correlation-based methods identify regulatory relationships by measuring the co-variation between TF expression, CRE accessibility, and gene expression across cells. Regression-based methods model gene expression as a function of multiple regulators simultaneously. Dynamical systems methods capture how gene expression changes over time as a function of regulatory inputs and stochastic effects. Finally, deep learning methods use neural networks to learn complex, non-linear relationships between TFs, CREs, and genes.

In addition to these four inference-based classes, literature-derived resources such as DoRothEA [27] and CollecTRI [28] provide pre-compiled TF–gene interaction networks assembled from curated databases, and serve as prior knowledge baselines.

2.2.1 Correlation-Based Methods

Correlation-based methods are the most direct approach to identifying regulatory relationships: a TF and a target gene are assumed to interact if their expression levels co-vary consistently across cells, and similarly a CRE is linked to a gene if its accessibility correlates with that gene’s expression. These methods are straightforward to interpret, but correlation alone has clear limitations. It cannot distinguish direct regulation from indirect co-regulation by a third factor, nor determine which variable is the regulator and which the target. Incorporating chromatin accessibility as an additional modality

¹The task of matching cells across two separately profiled datasets. Since the cells in each experiment are different, algorithms must infer which RNA cell corresponds to which ATAC cell based on shared biological signals.

²CellOracle, DeepMAPS, Dictys, FigR, GRaNIE, LINGER, Pando, SCENIC+.

³CollecTRI, DoRothEA, SCENIC.

partially addresses this, since requiring a TF to bind an accessible CRE near its target gene introduces a directional filter on top of the statistical association [3].

Pearson and Spearman correlations are the most commonly used, detecting linear and non-linear relationships respectively. As an example, FigR [29] (Figure 2.2) uses Spearman correlation to identify genes whose expression co-varies with many surrounding CREs, called domains of regulatory chromatin (DORCs). A DORC reflects a gene governed by multiple enhancers, each contributing to its regulation. TFs are then assigned to each DORC by scanning its accessible regions for known binding motifs⁴, linking a TF whenever its motif is enriched in the associated open chromatin.

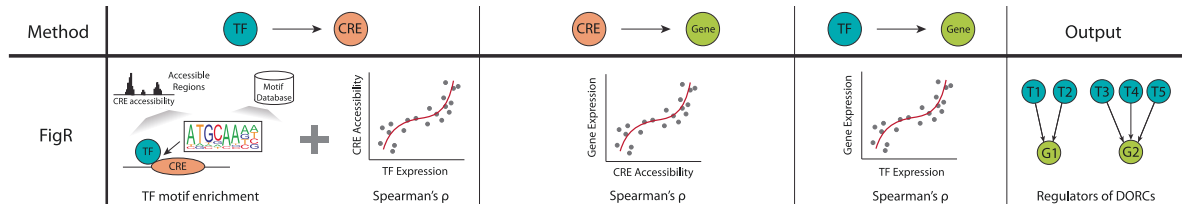


Figure 2.2: Schematic of FigR. From [3].

2.2.2 Regression-Based Methods

Regression-based methods model gene expression as a function of multiple regulators simultaneously. The expression of a target gene is treated as the response variable, predicted from the expression of candidate TFs and the accessibility of nearby CREs as predictor variables. Since gene expression is typically shaped by the combined effect of multiple regulators, this joint modelling captures regulatory relationships that pairwise correlation approaches cannot detect. The sign of each coefficient further indicates the direction of each regulatory interaction. Linear regression captures proportional relationships, while non-linear variants such as gradient boosting can model more complex regulatory dependencies [3].

SCENIC+ [30] (Figure 2.3) illustrates this approach. It first identifies candidate enhancers from scATAC-seq data using pycisTopic⁵, which extracts co-accessible genomic regions across cell types. TF binding motifs are then identified within these regions using pycisTarget⁶, producing TF cistromes: the set of regions bound by each TF.

GRNBoost2 [31] then scores both CRE–TG and TF–TG importance using gradient boosting regression. Combining TF cistromes with CRE–TG links yields candidate TF–CRE–TG triplets, and TF–TG importance scores serve as a co-expression filter to keep only the best links, producing the final enhancer-driven regulons (eRegulons).

⁴DNA sequences where transcription factors can bind to drive gene regulation, usually represented as nucleotide patterns [2].

⁵<https://github.com/aertslab/pycisTopic>

⁶<https://github.com/aertslab/pycistarget>

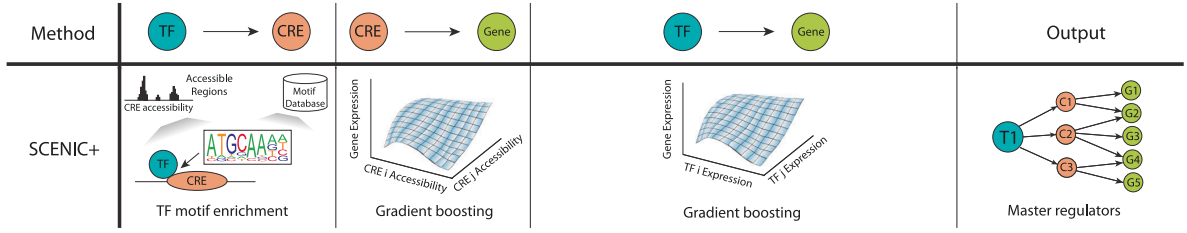


Figure 2.3: Schematic of SCENIC+. From [3].

2.2.3 Dynamical Systems Methods

Dynamic GRN inference methods model gene regulation as a process that evolves over time, capturing temporal changes in regulatory interactions and gene expression.

Dictys [32] (Figure 2.4) first identifies TF–CRE links using TF footprints⁷ and motif enrichment⁸ on chromatin accessibility data. Only TFs whose footprints fall near a given gene are retained as its candidate regulators, constraining the subsequent inference step. Gene expression is then modelled as a stochastic differential equation combining the regulatory contributions of these candidate TFs and a noise term accounting for biological stochasticity. The fitted coefficients quantify how the expression of each TF drives changes in target gene expression, yielding the final GRN.

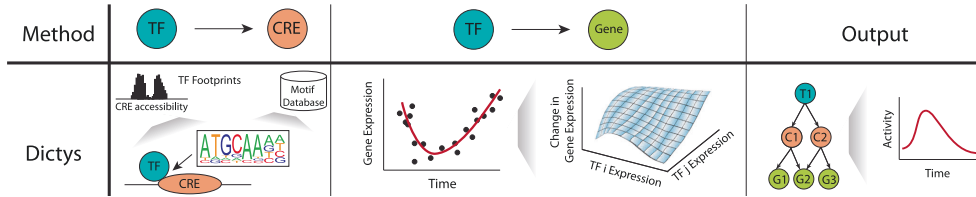


Figure 2.4: Schematic of Dictys. From [3].

2.2.4 Deep Learning Methods

Deep learning methods use neural networks that can learn complex, non-linear relationships between TFs, CREs, and genes directly from the data. We only provide a brief overview here since both methods will be detailed in the following chapters.

DeepMAPS [4] (Figure 2.5) processes gene expression and chromatin accessibility through a heterogeneous graph transformer, which jointly learns cell and gene embeddings. These embeddings are used to cluster cells and extract gene modules, from which TF–TG regulatory links are inferred to produce cell-type-specific GRNs.

⁷Precise regions of active TF binding within open chromatin peaks, identified by a localised drop in the accessibility signal where a bound TF physically shields the underlying DNA [33].

⁸The detection of overrepresented TF binding motifs within a set of regulatory regions of interest used to identify which TFs may be directly regulating those regions [34].

LINGER [35] trains one neural network per gene to predict its expression from TF expression and chromatin accessibility, first on atlas-scale bulk data and then refined on single-cell data. The network weights and Shapley values are then extracted to derive TF–CRE, CRE–TG and TF–TG regulatory scores.

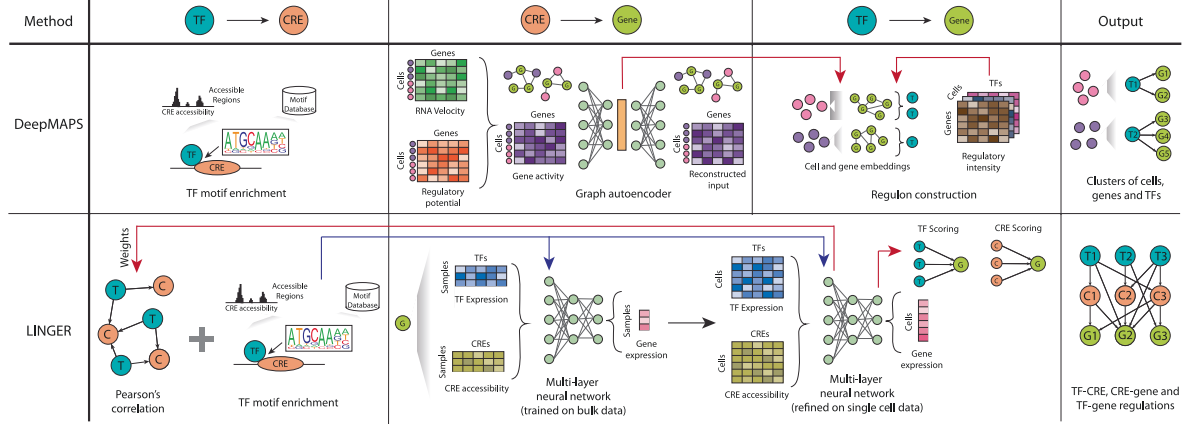


Figure 2.5: Schematic of DeepMAPS and LINGER. From [3].

3. LINGER

3.1 Introduction

LINGER (Lifelong Neural network for GEne Regulation) is a framework for inferring gene regulatory networks from single-cell multi-omics data [35]. It leverages atlas-scale bulk data from the ENCODE project [36] and TF binding motif databases as prior knowledge. LINGER first constructs a general GRN from bulk data, then refines it into a cell population GRN using the single-cell data, which can be further specialised to obtain cell type-specific GRNs.

3.2 Priors

To incorporate prior knowledge from atlas-scale external data, LINGER is pre-trained on 201 paired bulk RNA-seq and ATAC-seq samples from the ENCODE project, spanning diverse human cell types and tissues. For each target gene i , an individual neural network is trained to predict its expression from the expression of TFs and the chromatin accessibility of CREs within 1 Mb¹ of its transcription start site (TSS) (Figure 3.1). These networks are collectively referred to as BulkNN.

In addition, LINGER incorporates prior knowledge of TF–CRE binding from curated databases², encoded as a motif binding affinity matrix B where B_{tk} denotes the binding affinity of TF _{t} at CRE _{k} .

3.3 Preprocessing

LINGER takes two count matrices as input: $E \in \mathbb{R}^{I \times J}$, representing expression levels for I genes across J cells, and $O \in \mathbb{R}^{K \times J}$, representing chromatin accessibility for K peaks across the same J cells, together with cell type annotation. Because single-cell data is often noisy and highly sparse, cells are grouped into metacells that aggregate signals from multiple cells, making the signal more robust.

¹1 megabase (Mb) = 10⁶ base pairs. This window captures the CRE–gene interactions based on their genomic proximity.

²Including JASPAR [37], TRANSFAC [38], UniPROBE [39], and Taipale [40].

The pseudo-bulking works as follows. From E and O , principal component analysis (PCA) is performed to obtain two different low-dimensional embeddings for each cell (one derived from E and the other from O). These embeddings are combined to form a joint representation of each cell. A k -nearest neighbour (KNN) graph is then built to group similar cells together.

Metacells are constructed separately for each cell type by randomly sampling a set of cells as aggregation anchors. For each sampled anchor cell j , the corresponding metacell profile is obtained by aggregating the raw counts of all cells in its KNN neighbourhood. This yields the metacell matrices E' and O' , containing M metacells instead of the original J cells.

The rest of the preprocessing consists in creating inputs ready for neural network training. Peaks in O' are matched against reference peaks (a consensus chromatin accessibility map derived from the atlas-scale bulk data), inheriting the peak-to-gene associations already established for those reference peaks. Each matched peak is assigned to every gene whose TSS lies within 1 Mb of that peak. Genes in E' are filtered to keep only those present in the bulk reference, and TFs are restricted to those expressed in the data. This defines, for each gene, the set of TFs and CREs that will form the input nodes of its corresponding neural network (Figure 3.1).

3.4 Training

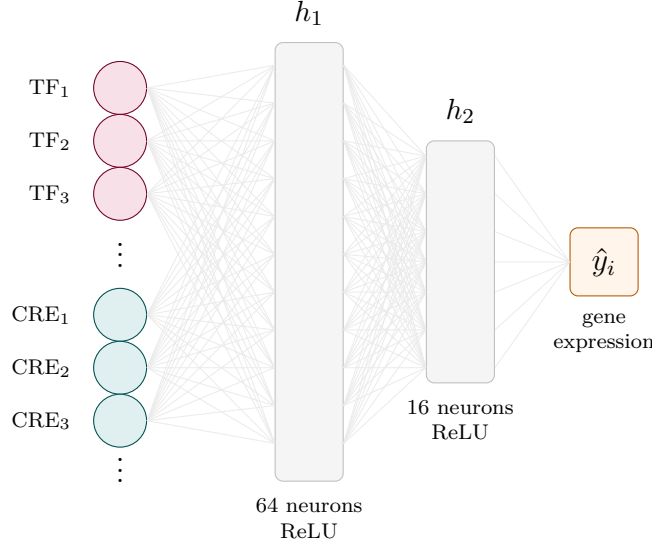
The cell population GRN is constructed by training one network per gene on single-cell data using elastic weight consolidation (EWC) [41], with the BulkNN as a regularisation target. Each gene network is randomly initialised and trained with a loss including an EWC penalty and a manifold regularisation term.

The EWC penalty constrains the first-layer weights to remain close to the BulkNN weights, preventing excessive drift and ensuring stable learning while allowing adaptation to single-cell-specific signals.

The manifold regularisation, built from the motif binding affinity matrix B , encourages the first-layer embeddings of TFs and CREs to be similar when their motif binding affinity is high, guiding the network toward regulatory modules in which TFs are co-clustered with the CREs they are known to bind.

After training, regulatory scores are extracted from the networks: the TF-CRE binding strength α is derived from the cosine similarity between first-layer embeddings of TFs and CREs, while the cis-regulatory strength β (CRE-TG) and trans-regulatory strength γ (TF-TG) are obtained by averaging the absolute Shapley values³ of CRE and TF inputs respectively, across all metacells.

³Shapley values quantify the contribution of each feature (TF and CRE) to each gene’s prediction [42]



α (TF $\rightarrow$ CRE): cosine similarity of embeddings of TFs and CREs in h_1
 β (CRE $\rightarrow$ TG): $|\overline{\text{SHAP}}|$ on CRE inputs
 γ (TF $\rightarrow$ TG): $|\overline{\text{SHAP}}|$ on TF inputs

Figure 3.1: **LINGER per-gene neural network architecture.** Each gene i has its own network taking TF expression and CRE chromatin accessibility within 1 Mb of its TSS as input. Inputs correspond to a single observation (a tissue sample for BulkNN pre-training, or a metacell for single-cell fine-tuning). After training, regulatory scores α , β and γ are extracted from the network weights and Shapley values $\star$.

3.5 Outputs

From the cell population GRN, cell type-specific GRNs are obtained by reweighting α , β , and γ using cell type-specific measurements, such as average chromatin accessibility $\bar{O}^{(c)}$ and average gene expression $\bar{E}^{(c)}$ within each cell type c . Equations (3.1), (3.2) and (3.3) are taken from the LINGER paper [35] and reflect the implementation in the released code⁴.

The TF–CRE binding score α_{tk} is updated by incorporating the mean accessibility of CRE _{k} , the mean expression of TF _{t} , and the motif binding affinity prior B_{tk} :

$$TFB_{tk}^{(c)} = \bar{O}_k^{(c)} \cdot (\alpha_{tk} + B_{tk}) \cdot \bar{E}_t^{(c)} \quad (3.1)$$

The cis-regulatory strength β_{ik} is similarly reweighted, with the addition of a distance decay term reflecting the diminishing influence of distal regulatory elements:

$$CRP_{ik}^{(c)} = \beta_{ik} \cdot \bar{O}_k^{(c)} \cdot \bar{E}_i^{(c)} \cdot e^{-\frac{d_{ik}}{d_0}} \quad (3.2)$$

⁴<https://github.com/Durenlab/LINGER/tree/main>

The trans-regulatory strength γ_{it} scales the cumulative effect of all CREs k within 1 Mb of TG_i 's TSS, the set S_i :

$$TRP_{it}^{(c)} = \gamma_{it} \sum_{k \in S_i} TFB_{tk}^{(c)} \cdot CRP_{ik}^{(c)} \quad (3.3)$$

A TF therefore strongly regulates a gene in cell type c if it binds accessible CREs near that gene and those CREs drive its expression.

3.6 Baseline version

LINGER also provides a baseline version that skips neural network training on single-cell data entirely. The population GRN is constructed directly from the pre-computed bulk prior matrices, and the single-cell data is used only as a feature mask (restricting to genes, TFs and reference peaks present in the user dataset). When deriving cell type-specific GRNs, the regulatory topology remains fully determined by the bulk prior, but the strengths are further adapted through cell type-specific scaling by $\bar{O}^{(c)}$ and $\bar{E}^{(c)}$.

3.7 Transcription Factor activity

From the trans-regulatory strength $\gamma \in \mathbb{R}^{I \times T}$ for T TFs, and the RNA expression matrix $E \in \mathbb{R}^{I \times J}$ for J cells, where I is the number of genes common to both matrices, LINGER infers TF activity at single-cell resolution by measuring how strongly each TF's target genes are expressed in each cell, relative to a null expectation.

The expression matrix E is first normalised such that each cell's expression values sum to one, then quantile-normalised: for each gene, cells are ranked by expression level, and each cell's expression is replaced by the mean expression across all genes at the same rank position. This makes expression levels comparable across genes, without distorting their relative patterns across cells.

An expected trans-regulatory matrix $E_{\text{exp}} \in \mathbb{R}^{I \times T}$ is constructed from γ as a null model, capturing the baseline co-activity expected from the marginal regulatory strength of each gene and TF alone, independently of their specific pairings. This allows the final activity score to reflect only the signal that exceeds what would be expected by chance.

This is achieved by summarising γ into gene-wise and TF-wise marginal regulatory strengths. The vector $\mathbf{r} \in \mathbb{R}^I$ denotes the gene-wise marginal regulatory strength obtained by summing regulatory weights across all TFs for each gene, and the vector $\mathbf{c} \in \mathbb{R}^T$ denotes the TF-wise marginal regulatory strength obtained by summing regulatory weights across all target genes for each TF. The total regulatory mass, defined as $Z = \sum_{i,t} \gamma_{it}$, is used to normalise the expected regulatory matrix:

$$E_{\text{exp}} = \frac{\mathbf{r}\mathbf{c}^\top}{Z} \quad (3.4)$$

The observed and expected TF activities are computed by projecting the regulatory network onto gene expression E , i.e., by aggregating the expression of TF target genes in each cell according to γ for the observed signal and E_{exp} for the expected signal.

$$\begin{cases} A_{obs} = \gamma^\top E \\ A_{exp} = (E_{exp})^\top E \end{cases} \in \mathbb{R}^{T \times J} \quad (3.5)$$

Finally, the TF activity matrix $A \in \mathbb{R}^{T \times J}$ is defined as a normalised enrichment that quantifies, for each transcription factor and each cell, the deviation of the observed regulatory signal from its expected value under the null model:

$$A = \frac{A_{obs} - A_{exp}}{A_{exp}} \quad (3.6)$$

3.8 Parallelisation

LINGER processes chromosomes sequentially throughout preprocessing, training and GRN construction. The latter refers to the post-training extraction of the three regulatory score matrices α , β and γ from the trained networks. Since chromosomes are independent from one another⁵, we parallelised the chromosome-level loops using Joblib⁶. Figure 3.2 summarises the speedups on a 24-CPU / 128 GB cluster (4 runs each, original vs. parallel) on the PBMC dataset [43] (Appendix A.2).

3.8.1 Preprocessing and training

In preprocessing, peaks are intersected per chromosome with the reference peak set to identify which user peaks overlap with known regulatory regions and inherit their gene associations. These independent intersections are dispatched in parallel with 8 workers; beyond 8, contention dominates and the speedup degrades. Preprocessing is light, so the gain is modest ($\times 1.8$). Training is parallelised across chromosomes with

$$n_{jobs} = \max\left(1, \min\left(\left\lfloor \frac{\text{RAM}_{free}}{5} \right\rfloor, n_{cpus}, 23\right)\right) \quad (3.7)$$

which reflects the ~ 5 GB per worker needed to hold the pre-trained networks, the available CPU count, and the fact that there are only 23 chromosomes. This yields the largest speedup ($\times 8.3$), at the cost of peak memory rising from 66 GB (original version) to 120 GB (parallel version).

⁵The per-chromosome prior files, pre-trained networks and Shapley values are all chromosome-stratified; the only cross-chromosome operation is the final concatenation of outputs.

⁶Python library for efficient parallel computing: <https://joblib.readthedocs.io/en/latest/parallel.html>

3.8.2 GRN reconstruction

All three steps detailed here are parallelised across chromosomes, each chromosome being processed independently before concatenating the results. For α , cosine similarities between TF and CRE first-layer embeddings are computed and aggregated into a TF-CRE matrix per chromosome.

This is the most expensive of the three steps, yielding a $\times 6.5$ speedup. For β and γ , the Shapley values previously computed for each gene-specific network are loaded per chromosome; for each gene, the contributions of CRE and TF inputs are read off to give β and γ respectively. Because the computationally heavy part has already occurred during training, the gains are smaller: $\times 4.6$ for β and $\times 5.5$ for γ .

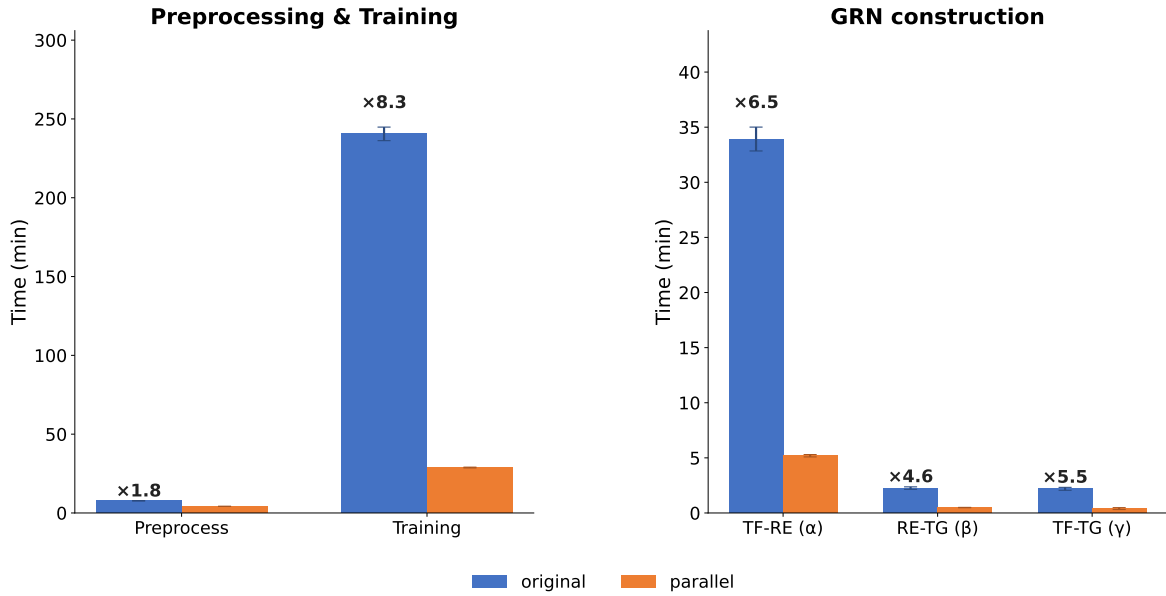


Figure 3.2: Benchmark of LINGER parallelisation on the PBMC dataset ^{*}.

All improvements described above have been integrated into the LINGER repository⁷.

⁷<https://github.com/Durenlab/LINGER/pull/75>

4. DeepMAPS

4.1 Introduction

DeepMAPS (Deep learning-based Multi-omics Analysis Platform for Single-cell data) is a framework for inferring cell-type-specific gene regulatory networks from single-cell multi-omics data [4]. It jointly processes gene expression and chromatin accessibility matrices through a heterogeneous graph transformer (HGT), which learns embeddings and attention scores for both genes and cells. Cells are first clustered using the HGT embeddings, and gene modules are then extracted per cluster. Transcription factor regulons¹ are finally inferred from these modules, producing GRNs that are inherently specific to each cell type.

Equations (4.1), (4.2), (4.4), (4.6) and (4.8) are taken from the DeepMAPS paper [4] and reflect the implementation in the released code².

4.2 Preprocessing

DeepMAPS takes as input a gene expression matrix $X^R \in \mathbb{R}^{I \times J}$ and a chromatin accessibility matrix $X^A \in \mathbb{R}^{K \times J}$. X^R stores the expression of I genes across J cells, while X^A stores the accessibility of K chromatin peaks across the same cells. The HGT trains on a single gene activity score (GAS) matrix $X^G \in \mathbb{R}^{I \times J}$, derived from the two matrices above. X^G is constructed as follows.

First, peaks in X^A are linked to their neighbouring genes through a gene-peak regulatory potential matrix $W \in \mathbb{R}^{I \times K}$, whose entries w_{ik} encode the influence of peak k on gene i based on their genomic proximity. The gene regulatory potential matrix $X^{A'} \in \mathbb{R}^{I \times J}$ is then obtained by weighting each peak’s contribution by its accessibility and summing over all peaks near gene i :

$$x_{ij}^{A'} = \sum_k w_{ik} \cdot x_{kj}^A \quad (4.1)$$

The GAS matrix X^G is constructed by weighting $X^{A'}$ and X^R according to RNA velocity, which reflects the ratio of unspliced to spliced mRNA. Unspliced mRNA is a nascent transcript that has not yet been processed, while spliced mRNA is the mature form ready for translation; a high ratio of unspliced to spliced therefore indicates that

¹group of genes regulated by the same transcription factor.

²<https://github.com/OSU-BMBL/deepmaps>

a gene is being actively transcribed [44]. If the velocity is positive, the gene has a high level of unspliced RNA, indicating active transcription; chromatin accessibility is therefore more informative and is upweighted. If the velocity is negative, transcription is declining and gene expression is more relevant; chromatin accessibility is consequently downweighted.

4.3 Training

Hereafter, X^G is referred to as $X \in \mathbb{R}^{I \times J}$: the integrated matrix containing combined activity of gene i in cell j .

4.3.1 Initial embeddings

In X , cells and genes lie in high-dimensional spaces of dimension I and J , respectively. Two MLP autoencoders³ (AE) are used to learn their initial embeddings: the cell autoencoder takes each column of X (the expression profile of all I genes in each cell) as input, and the gene autoencoder takes each row (the expression of each gene across all J cells).

Each autoencoder is a feedforward neural network that compresses its input vector x into a latent embedding $z \in \mathbb{R}^{256}$ via an encoder, then reconstructs $\hat{x}$ from z via a decoder. The network is trained by minimising $\|x - \hat{x}\|^2$, forcing z to retain only the most informative structure of the input.

4.3.2 Heterogeneous graph transformer model

From the integrated matrix X , we construct a bipartite heterogeneous graph G in which the nodes represent either cells or genes. A link between cell j and gene i exists whenever the gene is active in the cell, i.e. $x_{ij} > 0$. The initial node embeddings $\mathcal{H}^0 \in \mathbb{R}^{N \times d}$ are learned with the two AEs described above, where N is the total number of nodes (cells and genes) and d is the embedding dimension.

The graph G is then passed to pyHGT [45], a graph neural network framework designed for heterogeneous graphs. Each node updates its embedding by aggregating information from its neighbours, weighting the contribution of each neighbour with attention scores. The model outputs final embeddings for both cells and genes, along with attention scores on edges that quantify the importance of each gene–cell interaction.

4.3.3 Embedding update process

The model is built from L stacked HGT layers. Figure 4.1 illustrates a single such layer. At each layer, every node is treated in turn as a target (red circle) and its embedding is updated by aggregating messages from its 1-hop neighbours, which act as sources (black circles).

³A Multi-Layer Perceptron (MLP) is a feedforward neural network of stacked linear layers with non-linear activations. The cell and gene autoencoders use symmetric architectures $I \rightarrow 512 \rightarrow 256 \rightarrow 512 \rightarrow I$ and $J \rightarrow 512 \rightarrow 256 \rightarrow 512 \rightarrow J$, respectively, with ReLU activations throughout.

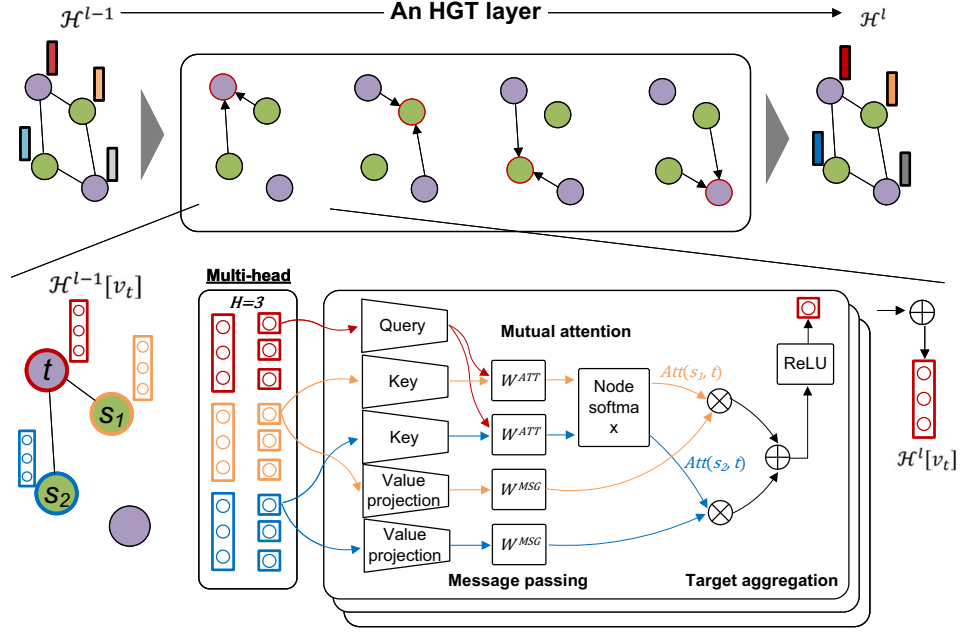


Figure 4.1: Single HGT layer: per-node embedding update via multi-head attention. From [4].

The upper panel shows that this process is applied to every node in the graph, producing updated embeddings $\mathcal{H}^l$ from the previous layer's embeddings $\mathcal{H}^{l-1}$.

The bottom panel details the update for a single target node t with two source neighbours s_1 and s_2 . Each node carries an embedding from the previous layer $\mathcal{H}^{l-1}$, represented as coloured rectangles. The model uses multi-head attention with H heads, each in its own lower-dimensional subspace of size $d_h = d/H$, allowing different heads to capture different interaction patterns simultaneously.

For each head, three vectors are derived from the current embeddings through learned linear projections: a query from the target node, and a key and value from each source node. These projections are learned weight matrices that map node embeddings into the attention subspace of dimension d_h . They are node-type specific, so cells and genes use different matrices, allowing each modality to be encoded in its own way. The key of each source and the query of the target are compared via a scaled dot product, modulated by W^{ATT} , a learned matrix that makes the score sensitive to the types of both nodes. The resulting scores are normalised across all neighbours with a softmax, yielding the attention coefficients $Att(s_1, t)$ and $Att(s_2, t)$.

Each source then sends a message to the target by passing its value through a message matrix W^{MSG} , which adapts the message to the type of the target node, before weighting it by the corresponding attention coefficient. The weighted messages from all sources are summed, concatenated across all H heads, passed through a ReLU non-linearity, and finally combined with the target node's previous embedding $\mathcal{H}^{l-1}[t]$, yielding the updated embedding $\mathcal{H}^l[t] \in \mathbb{R}^d$.

4.3.4 Attention scores

After the final layer L , the per-head attention coefficients $\text{Att}^h(i, j)$ associated with each gene–cell edge are aggregated into a single scalar by computing their ℓ_2 norm across heads:

$$a_{i,j} = \sqrt{\sum_{h=1}^H \left(\text{Att}^h(i, j)\right)^2} \quad (4.2)$$

These scores $a_{i,j}$ quantify the importance of gene i to cell j and are reused downstream when building cell-type-specific gene modules.

4.3.5 Loss

The HGT is trained following the same autoencoder principle described in Section 4.3.1: the input is the integrated matrix X , which the model attempts to reconstruct. The encoder produces final embeddings $\mathcal{H}^L$ through L transformer layers, and the decoder reconstructs X via an inner product between the final embeddings of gene i and cell j :

$$\hat{x}_{ij} = \mathcal{H}^L[i]^\top \mathcal{H}^L[j] \quad (4.3)$$

where $\hat{x}_{ij}$ is the predicted activity of gene i in cell j . The model is trained by minimising the Kullback–Leibler divergence, a measure of how much one probability distribution differs from another [46], between the reconstructed and original matrices.

Both matrices are first normalised with a softmax along the cell dimension, so that for each gene, activity scores across cells form a probability distribution summing to one. The KL divergence then compares how well the predicted distribution of activity across cells matches the observed one:

$$\text{loss} = \text{KL}(\text{softmax}(\hat{X}), \text{softmax}(X)) \quad (4.4)$$

4.4 Outputs

4.4.1 Cell clustering

The final cell embeddings produced by the HGT are used for cell clustering. A k -nearest-neighbour (KNN) graph is first constructed in the HGT embedding space using cosine distance, connecting each cell to its $k = 20$ closest neighbours. A shared nearest-neighbour (SNN) graph is then derived from the KNN graph, by weighting each edge between cells j_1 and j_2 according to the size of their shared neighbourhood:

$$w_{j_1, j_2} = \frac{|\mathcal{N}(j_1) \cap \mathcal{N}(j_2)|}{|\mathcal{N}(j_1) \cup \mathcal{N}(j_2)|} \quad (4.5)$$

Biologically similar cells (for example cells of the same type) tend to share many neighbours, so this step sharpens cluster boundaries compared to a KNN graph. The Louvain algorithm [47] is then applied to the SNN graph, yielding the final cell clusters used in the subsequent step.

4.4.2 Functional gene modules

This step builds cell-type-specific gene modules by solving a Steiner Forest Problem (SFP) on a graph combining HGT attention scores and gene embeddings. Cells and genes are its nodes, and two types of edges are built.

Gene-cell edges are derived from the attention scores: for each cell j , only genes i with strong attention to that cell are kept, i.e. $a_{ij} > \mu_j + \sigma_j$, where μ_j and σ_j are the mean and standard deviation of the attention scores received by cell j from all its neighbouring genes. Gene-gene edges are derived from the gene embeddings: two genes are connected if the Pearson correlation between their embeddings exceeds 0.5. Edge weights are then inverted so that strong biological signal corresponds to low cost.

The goal is to identify, for each cluster, a small set of genes collectively relevant to all its cells. This is achieved by seeking a minimum-cost subgraph that connects all cells within each cluster. Cells of cluster z are terminals (must-cover nodes) and genes are Steiner nodes (optional bridges), retained only if they help connect terminals at lower cost. A gene surviving in cluster z 's tree necessarily bridges at least two cells; genes connecting to only one cell are pruned. The genes retained in each cluster's tree define its co-active module co_z .

Since the SFP is NP-hard, an approximation is used in two steps. First, the gene-cell edges are sparsified by iterative maximum bipartite matching on the original (non-inverted) weights: in each iteration, the strongest available gene-cell links are selected such that no gene or cell appears more than once in that iteration's matching. Covered cells leave the pool and the process repeats on the remaining uncovered cells. As a result, each cell accumulates exactly one gene-cell edge, while the same gene can be matched to different cells across iterations, accumulating several gene-cell edges. Any cell still uncovered after all rounds is assigned its single strongest gene-cell edge as a fallback. Edge weights are then inverted and merged with the gene-gene edges.

Second, for each cluster, a minimum spanning tree is computed on the induced subgraph consisting of the cluster's cells, their associated genes from the sparsified gene-cell edge set, and all gene-gene edges connecting those associated genes.

This yields one co-active module co_z per cluster, comprising genes with high attention scores towards that cluster's cells and high mutual correlation in embedding space.

4.4.3 From gene modules to regulons

For each cluster z , its gene module co_z is used to infer a set of TF regulons: groups of genes from co_z controlled by the same transcription factor. This produces, for each cluster, a list of TFs together with their target genes, forming the TF-TG links that constitute the cell-type-specific GRN. This inference is performed in three steps.

First, the gene-peak regulatory potential matrix W , where w_{ik} denotes the regulatory potential weight of peak k for gene i , is restricted to genes whose promoter region contains at least one accessible ATAC peak. To this end, for each gene in W , the genomic window spanning 2 kb upstream of the gene body is derived from the Ensembl annotation database [48] and intersected with the ATAC peak coordinates in

the chromatin accessibility matrix X^A . Genes with no overlapping peak are removed, since a transcription factor can only regulate a gene if its promoter region is accessible. For simplicity, we continue to denote the restricted matrix by W .

Second, candidate TFs for each cluster are identified by linking the genes of co_z to their accessible peaks and querying which TFs bind those peaks. For each gene i in co_z , the peaks k with a non-zero regulatory potential weight w_{ik} in W are retained. These peaks are then intersected with JASPAR’s database: a genome-wide catalogue of transcription factor binding sites that indicates which TFs bind to specific genomic regions. Whenever one of these retained peaks overlaps with a JASPAR binding site, the corresponding (peak, TF) link is recorded in a peak-by-TF binding affinity matrix B , where b_{kt} represents the binding affinity of TF t at peak k .

The gene–peak regulatory potential matrix W and the peak–TF binding matrix B are then combined to attribute TF binding to genes:

$$\text{geneTF}_{it} = \sum_k w_{ik} \cdot b_{kt} \quad (4.6)$$

so that it is non-zero if at least one peak k lies within the promoter region of gene i and contains a binding site for TF t . The regulon $R_{t,z}$ of TF t in cluster z is then defined as the subset of genes in co_z for which this condition holds:

$$R_{t,z} = \{i \in co_z \mid \text{geneTF}_{it} > 0\} \quad (4.7)$$

Third, the candidate TFs are refined using LISA, an external method that ranks TFs by how well their known regulatory profiles match the gene list co_z of the cluster. Among all candidate regulons $R_{t,z}$, only those whose TF passes both the JASPAR motif overlap and the LISA filter are retained as final regulons.

4.4.4 Cell-type-specific GRNs

For each TF t and each TG i in its regulon $R_{t,z}$, a regulatory intensity score $s_{ti|z}$ quantifies how strongly the regulation actually manifests in cluster z . It is obtained by averaging, across the cluster’s cells, the per-cell regulatory intensity score:

$$s_{ti|z}^j = \sum_k \tilde{x}_{kj}^A \cdot w_{ik} \cdot b_{kt} \quad (4.8)$$

where $\tilde{x}_{kj}^A \in \{0, 1\}$ is the binarised entry of the chromatin accessibility matrix X^A , extending the conditions defining geneTF_{it} by additionally requiring peak k to be open in cell j .

Target genes with zero cluster-mean intensity are dropped from $R_{t,z}$, leaving only those whose chromatin path is accessible in at least some cells of the cluster. The retained TF–TG edges, together with their $s_{ti|z}$ scores, define the cluster-specific gene regulatory networks.

Each edge is further annotated with its supporting cis-regulatory elements (Equation 4.9), the set of peaks k simultaneously linked to gene i and bound by TF t , yielding the final TF–CRE–TG triplets.

$$\mathcal{C}_{ti} = \{k : w_{ik} > 0 \text{ and } b_{kt} > 0\} \quad (4.9)$$

Because a gene is included in a regulon only if at least one peak already connects it to the TF, every retained edge is guaranteed to have at least one supporting CRE.

4.5 Issues Encountered with the functional gene modules

The functional gene module step aims at finding cell-type-specific gene modules by solving an approximation of the Steiner Forest Problem on a graph combining HGT gene–cell attention scores and gene–gene embedding correlations. Running this step on the PBMC dataset produced modules that were both biologically unexpected and largely redundant across clusters, for reasons that appear to be rooted in the structure of the graph itself.

The recovered gene modules contained exclusively Z-family genes (mostly Zinc Finger genes *ZNF*, *ZSCAN*, and *ZSWIM*), which are not expected markers for blood cell types. Furthermore, the 10 recovered cluster modules were near-identical: the two largest clusters ($n = 111$ and $n = 105$ genes) shared 85 genes, corresponding to 77% and 81% of their respective gene sets.

Even the smallest cluster ($n = 41$ genes) had 80% of its genes appear in the two larger clusters. While cluster-specific modules should ideally contain genes active in that cell type, the near-total overlap observed here shows that the algorithm is recovering the same background *ZNF* block rather than biologically meaningful, cluster-discriminating genes.

Two structural properties of the graph construction may explain this behaviour. First, the gene–gene edge set is dramatically denser than the gene–cell edge set. With a Pearson correlation threshold of 0.5 applied to the HGT gene embeddings, 73% of all gene pairs are connected, producing approximately 79.6 million gene–gene edges. By contrast, the attention threshold retains only 143,860 gene–cell edges out of 132 million possible pairs (0.1%), covering only 254 unique genes (a ratio of roughly 550 gene–gene edges for every gene–cell edge).

Since edge weights are inverted before the minimum spanning tree step, such that high correlation translates to low cost, highly correlated genes become cheap intermediate nodes for the Steiner forest approximation. Within this already dense graph, *ZNF*-family genes form a particularly tight clique: 90% of *ZNF*–*ZNF* pairs exceed the threshold against 73% for other gene pairs, and their mean pairwise embedding correlation (0.77) is notably higher than that of other genes (0.65). As a result, the *ZNF* block constitutes a cheap highway through the graph, which the Steiner forest preferentially routes through regardless of the cluster being processed.

A possible solution would be to raise the gene-gene correlation threshold, so as to reduce the density of the gene-gene clique, while simultaneously lowering the attention threshold to broaden the pool of candidate gene-cell edges. Although each cell still accumulates exactly one gene-cell edge after matching, a larger and more diverse candidate pool might reduce the tendency of cells to converge on the same small set of 254 candidates, potentially giving the minimum spanning tree more cluster-discriminating signal and less reason to route through the cheap ZNF backbone. Whether this combined adjustment would be sufficient to recover biologically meaningful modules remains to be verified empirically.

5. GRETA

5.1 Introduction

GRETA (Gene Regulatory nETwork Analysis) is a free open-source framework designed to compare and evaluate methods for inferring gene regulatory networks from single-cell multi-omics data [5]. Comparing and evaluating the quality of inferred GRNs is challenging due to the lack of a true ground reference, as no complete and experimentally validated map of all TF–TG regulatory interactions exists. To address this, GRETA compares GRNs using similarity and stability scores, and evaluates them through three categories of metrics, described in Section 5.4.

Similarity measures the agreement between GRNs inferred from the same dataset by different methods, using the overlap coefficient at the TF, target gene, and edge levels to quantify how much their outputs converge. Stability assesses the robustness of each method’s output across repeated runs and under progressive downsampling of cells or features, capturing how much the inferred networks change with randomness and data coverage.

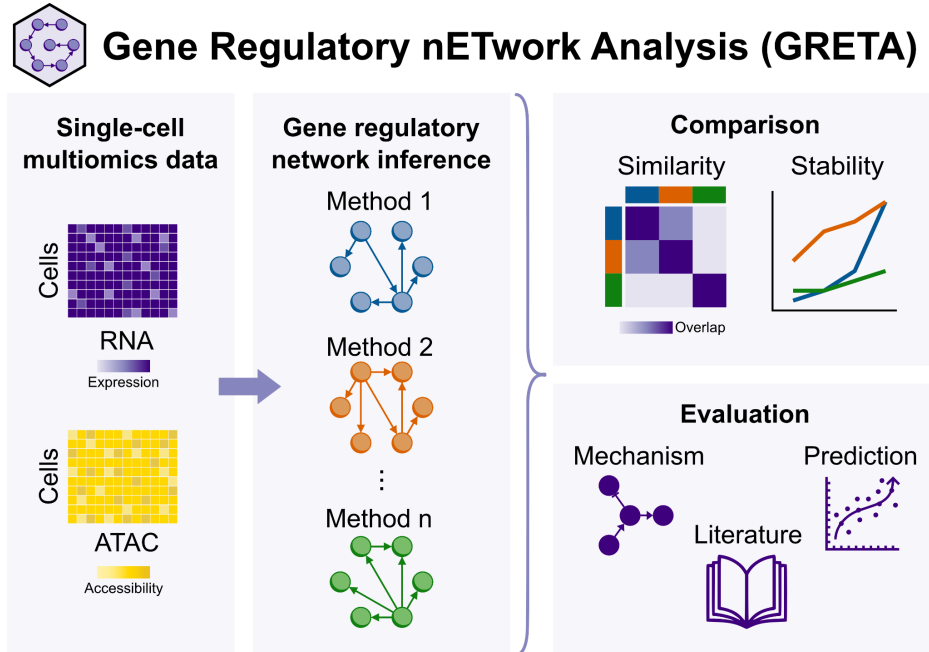


Figure 5.1: Graphical abstract of GRETA. From [5]

5.2 Adding a new dataset

GRETA is implemented as a Snakemake pipeline [49], in which rules break the workflow into smaller steps. Each rule defines a set of input files and produces a set of output files. Snakemake constructs a directed acyclic graph (DAG) of rules based on their dependencies.

To illustrate this workflow, we describe the steps required to integrate the *Single-cell multiome of human RPE and choroid* dataset [50] into GRETA. The dataset is publicly available on the Gene Expression Omnibus (GEO), a public repository for genomics data [51]. The resulting DAG is shown in Figure 5.2.

The pipeline starts by downloading two Singularity images using the `dwn_image` rule. These images provide reproducible environments for all downstream steps. In parallel, Ensembl gene IDs [52] are retrieved by the `gen_gid_ensembl` rule for later gene validation.

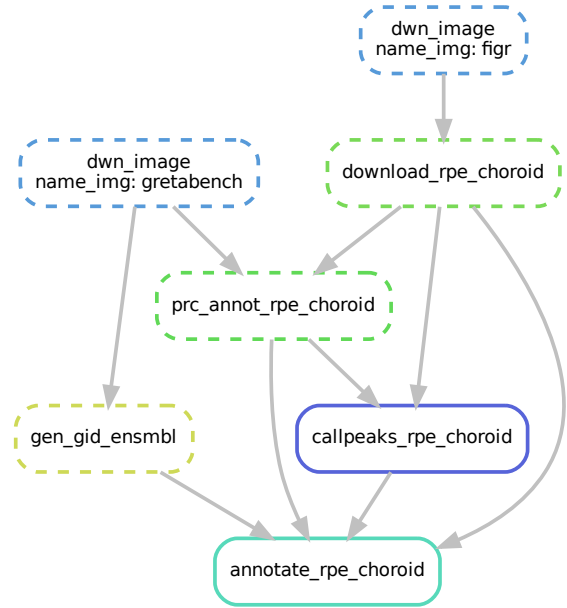


Figure 5.2: DAG for dataset integration

The pipeline then proceeds as follows. First, the raw data is downloaded from GEO using the `download_rpe_choroid` rule, including the RNA expression matrix, containing raw counts indexed by cell barcodes and gene names, along with per-cell-type ATAC fragment files.

Since each fragment file corresponds to a single cell type, cell barcodes can be directly mapped to their cell type based on which file they originate from. This mapping is then matched against the RNA barcodes to generate the final barcode to cell type annotation using the `prc_annot_rpe_choroid` rule.

Next, chromatin accessibility peaks are identified from the fragment files using MACS3 [17]. In the `callpeaks_rpe_choroid` rule, fragments are pooled per cell type and regions of significantly enriched accessibility (peaks) are recovered to construct the chromatin accessibility matrix.

Finally, both modalities are merged into a single multi-modal object through the `annotate_rpe_choroid` rule. During this step, genes are validated against Ensembl gene IDs to remove invalid gene symbols.

All methods in GRETA take as input datasets processed as described above, resulting in a single multi-modal MuData object [53]. This container stores multiple modalities together, where each modality is represented as an AnnData object [54].

An AnnData object centres around a matrix X of shape $\text{cells} \times \text{features}$, where features correspond to genes in the RNA modality and peaks in the ATAC modality. Rows and columns are annotated by *obs* and *var* respectively, storing cell-level metadata such as cell type and batch, and feature-level metadata such as gene IDs or peak

coordinates. X typically contains normalised and processed counts, while *layers* allows storing additional versions of the matrix, such as the raw counts. All modalities in the MuData object share the same cell observation index, ensuring that each row corresponds to the same cell across modalities.

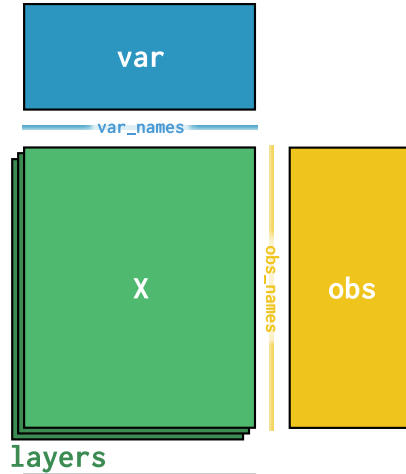


Figure 5.3: AnnData schema representation. From [6].

5.3 Adding new methods

GRETA aims to benchmark GRN inference methods under a unified setting. First, every method takes as input a MuData object as described in Section 5.2, from which the raw counts are systematically retrieved, since each method may apply its own preprocessing and normalisation.

Second, all methods integrated in GRETA are expected to produce a list of weighted TF–TG interactions at the cell-population level rather than per cell type, ensuring that outputs are directly comparable across methods. This choice is motivated by the fact that not all methods natively produce cell-type-specific GRNs, and that ground-truth databases with cell-type resolution remain rare, making a fair cell-type-level comparison across methods currently impractical.

Some methods go further and produce TF–CRE–TG triplets together with their scores, which provide a more complete GRN and make it possible to compute certain CRE-specific metrics.

Finally, every method must be packaged in a Singularity image to ensure reproducibility.

5.3.1 Decoupling

Methods can be integrated in two ways: a decoupled version, which splits the GRN inference process into four steps, and an original version, which keeps the inference pipeline unified.

GRN construction can be split into four inference steps, described in Figure 5.4.

1. [pre] Preprocessing: identify candidate TFs, CREs, and TGs
2. [p2g] Peak-to-gene mapping: assign peaks to nearby genes based on genomic distance between each peak and the gene TSS.
3. [tfb] TF binding: link TFs to CREs using motif-matching scores.
4. [mdl] Modelling: infer final TF–TG interactions.

Each step from one method can be plugged into another method’s pipeline, enabling the construction of GRNs composed of inference steps from different methods. In the GRETA paper, it was shown that most original methods outperform all mixed combinations¹, suggesting that mixing inference steps does not lead to improved performance.

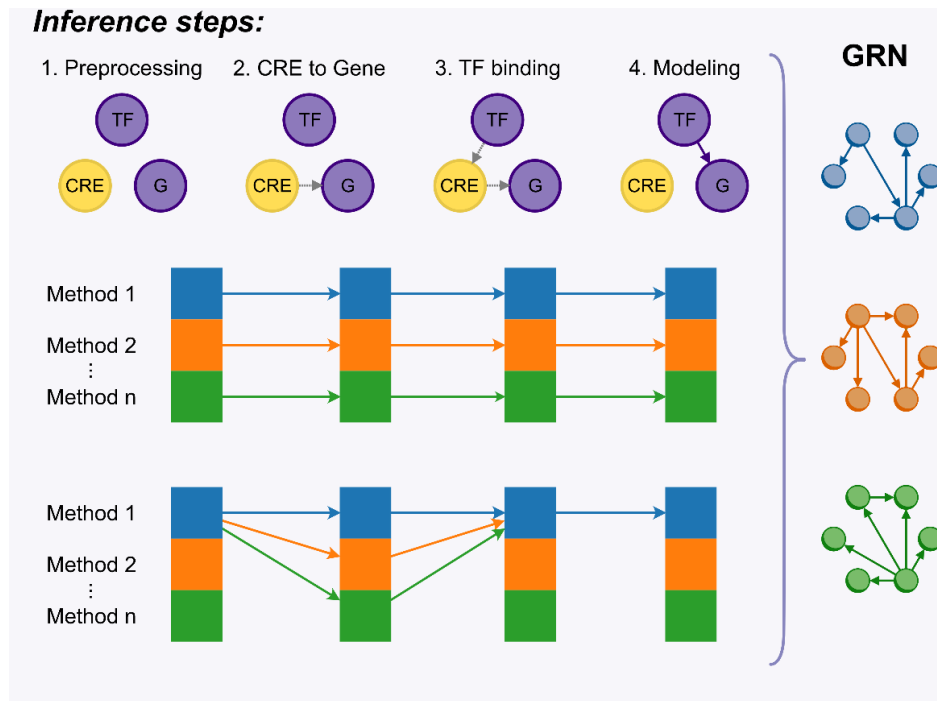


Figure 5.4: GRN inference steps. From [5]

Deep learning-based GRN inference methods are not decouplable, as training produces all outputs simultaneously. Consequently, only the original implementations of LINGER and DeepMAPS were integrated into GRETA.

LINGER needs its pre-trained neural networks to create the final GRN and only extracts the results of the different steps once the training is over. DeepMAPS needs outputs of the heterogeneous graph transformer model (cell and gene embeddings as well as their attention scores) to infer TF regulons and build the GRNs.

¹except Pando [55], SCENIC [56], and the random baseline (Section 5.3.4).

5.3.2 LINGER integration

We illustrate the integration of LINGER² using its Snakemake DAG when running on the PBMC dataset (Figure 5.5). The rules producing `annotate_pbmc10k` correspond to the dataset processing pipeline described in Section 5.2. The LINGER-specific part of the DAG consists of three additional steps: the download of prior knowledge, the preparation of the input data, and the GRN inference itself.

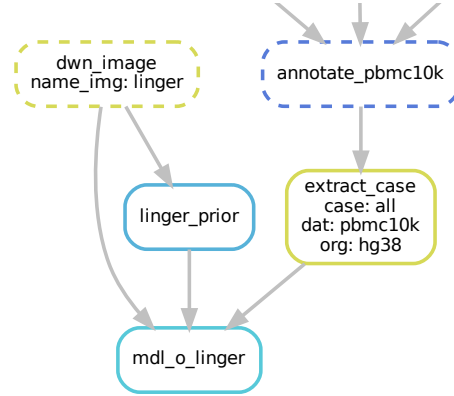


Figure 5.5: DAG for LINGER integration

The `linger_prior` rule downloads the prior knowledge required by LINGER: the pre-trained neural networks on bulk data and the prior binding affinity matrix B described in Section 3.2. The `extract_case` rule then applies the common preprocessing steps described in Section 1.3, along with additional steps such as dimensionality reduction and visualisation embeddings. Finally, the `mdl_o_linger` rule performs the GRN inference. LINGER can run in original mode (no parallelisation), baseline mode (Section 3.6), or parallel mode (with the optimisations described in Section 3.8).

LINGER produces three outputs from which TF–CRE–TG triplets can be reconstructed: α (TF–CRE), β (CRE–TG) and γ (TF–TG). Each matrix is first independently filtered to its top 10% scores to reduce the combinatorial explosion arising from the subsequent joins.

First, we join α and β along their shared CRE dimension to form $TF_t-CRE_k-TG_i$ triplets. Next, we perform a second join on the shared (TF, TG) dimension to incorporate γ , discarding any triplets that lack independent TF–TG support. The score of each remaining triplet is then defined as the product of its three component scores:

$$\text{score}(t, k, i) = \alpha_{tk} \cdot \beta_{ik} \cdot \gamma_{ti} \quad (5.1)$$

While γ alone could have been used as a direct source of TF–TG interactions, the validity of this triplet-based construction against that simpler alternative is evaluated in Section 6.1.

5.3.3 DeepMAPS integration

We integrated DeepMAPS³ into GRETA following the same structure as LINGER. A Singularity image was built from the DeepMAPS repository [57], and the method was run on the same PBMC dataset.

²<https://github.com/saezlab/greta/pull/54>

³Since DeepMAPS could not be totally reproduced, we did not create a dedicated pull request but its integration code can be found here: <https://github.com/emvg/greta/tree/hercules>.

Since DeepMAPS is implemented primarily in R (with the exception of HGT model training, which is in Python), the MuData object produced by `extract_case` was converted into a Seurat object [58], a commonly used R data structure for single-cell genomics, before being passed as input to the pipeline.

Unlike most methods, DeepMAPS natively produces cell-type-specific GRNs as $TF_t-CRE_k-TG_i$ triplets. For each cluster z , the inferred regulons $R_{t,z}$ directly define TF-TG edges, each annotated with its supporting cis-regulatory elements $\mathcal{C}_{ti}$ (as defined in Section 4.4.4) and a regulatory intensity score $s_{ti|z}$. To obtain a cell-population-level GRN, triplets were collected across all clusters, and for each unique $TF_t-CRE_k-TG_i$ triplet the maximum $s_{ti|z}$ over all z was retained as the final edge score.

Deviations from the original implementation

Two adaptations from the original implementation were necessary. The first concerns the construction of the gene activity score matrix, which originally relies on RNA velocity, and the second concerns the gene module construction step, where the SFP branch produced unsatisfactory results. Both are described below.

Gene activity score computation. As described in Section 4.2, the GAS matrix is originally built by weighting the RNA and ATAC modalities together using an RNA velocity matrix, a gene-by-cell matrix in which each entry encodes the rate of change of a gene’s expression in a given cell.

Estimating the velocity requires separate counts for spliced and unspliced mRNA, whereas the standard RNA expression matrix used as input to GRETA only stores total counts per gene, with no distinction between the two. Recovering them would require going back to the raw sequencing reads and processing them with a velocity-aware tool that separately counts spliced and unspliced mRNA, before passing the resulting counts to a velocity estimation method such as scVelo [59].

GRETA is meant to be applied to a wide range of datasets, and requiring users to provide raw sequencing reads for every new dataset would add a substantial preprocessing burden, change the input requirements of the pipeline, and prevent this step from being fully automated within the workflow.

We therefore used the weighted nearest neighbours (WNN) alternative to replace the velocity-based weights for integrating the RNA and ATAC modalities [60]. This approach constructs a weighted nearest neighbour graph by identifying, for each cell, its nearest neighbours based on a weighted combination of the two modalities.

This results in a pair of per-cell weights, one for RNA and one for ATAC, summing to one, which are then used in the GAS construction. This preserves the original intent of weighting RNA expression and chromatin accessibility by their relative informativeness in each cell, without requiring spliced/unspliced quantification.

Gene module construction. As described in Section 4.4.2, the construction of cell-type-specific gene modules in DeepMAPS relies on solving an approximation of the Steiner Forest Problem (SFP) on a graph combining HGT attention scores and gene embeddings.

In practice, the SFP branch produced biologically unexpected and largely redundant modules, for the structural reasons detailed in Section 4.5. We therefore fell back to the alternative branch, which defines gene modules directly from the HGT attention scores. Unlike the SFP branch, which applies the attention threshold at the cell level, this branch first averages the attention scores a_{ij} over all cells j belonging to cluster z , yielding a per-gene-per-cluster score $\bar{a}_{iz}$. The threshold is then applied at the cluster level: a gene i is retained in module co_z if:

$$\bar{a}_{iz} > \mu_z + \sigma_z \quad (5.2)$$

where μ_z and σ_z are the mean and standard deviation of $\bar{a}_{iz}$ across all genes within cluster z .

This approach produced 11 gene modules, with sizes ranging from 27 to 99 genes, and yielded more biologically interpretable results than the SFP branch. Cell-type marker genes were recovered across modules, such as *NKG7* [61] and *GZMA* [62], which are well-established markers of NK cells and cytotoxic T lymphocytes, both present in the PBMC dataset.

Modules showed an average pairwise Jaccard similarity⁴ of 0.17, with values ranging from 0.05 to 0.50, suggesting that modules are mostly cluster-specific, though some overlap remains, with 7 genes shared across all clusters.

5.3.4 Random baseline

A method’s performance is only informative relative to what could be obtained by chance. To this end, GRETA constructs a topology-preserving random network that matches key statistical properties of inferred GRNs, such as the number of CREs per gene, the number of TFs per CRE, the TF-to-gene ratio, and the genomic window constraints, while removing all biological signal.

To achieve this, gene, CRE, TF, and cell labels in the input MuData object are randomly permuted, breaking all true biological correspondences while leaving the expression and accessibility matrices unchanged.

A subset of genes is then sampled uniformly at random, and each gene is assigned a random number of CREs drawn from peaks located within a ± 250 kb window around the transcription start site. TFs are subsequently assigned randomly to CREs using the Lambert human TF catalogue [63], producing synthetic TF–CRE–gene triplets. Finally, the network is filtered to match realistic regulon statistics by enforcing a TF-to-gene ratio of 10% and removing TFs with fewer than five targets.

The resulting graph preserves the global topology of a GRN while containing no true regulatory relationships, providing a lower bound for evaluation.

⁴Jaccard similarity: $J(A, B) = \frac{|A \cap B|}{|A \cup B|}$.

5.4 Evaluation metrics

GRETA uses three metrics to evaluate GRNs: mechanistic metrics capturing causal relationships, predictive metrics reflecting network topology, and literature-based metrics incorporating prior biological knowledge.

For each metric, precision $P = \frac{TP}{TP+FP}$ and recall $R = \frac{TP}{TP+FN}$ are computed according to metric-specific definitions of true positives, false positives, and false negatives, and summarised into a precision-weighted F score:

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{\beta^2 \cdot P + R} \quad (5.3)$$

Since GRN inference methods try to capture the most confident interactions (min. FP) rather than all possible ones in ground truth (min. FN), precision is weighted ten times more than recall ($\beta = 0.1$), favouring quality over quantity.

Ground-truth databases are inherently incomplete and capture only a fraction of true regulatory interactions. In this context, missing a true interaction (FN) is less penalised than predicting an interaction that is not real (FP) because the ground truth itself cannot be fully trusted as a complete reference.

GRN inference methods can be run on many different datasets. GRETA provides a whitelist of tissue and cell-type labels that are considered biologically relevant for each dataset. When benchmarking a method run on a given dataset, the ground-truth databases are filtered to keep only experiments from tissues and cell types that match that biological context.

For instance, when evaluating a method trained on the peripheral blood dataset, only ground-truth experiments originating from blood, immune, or hematopoietic cell types are retained. Experiments from brain or lung cell lines would be excluded as irrelevant. The databases used as ground truth and the definition of evaluation metrics can be found in Appendix A.3 and A.4.

5.4.1 Mechanistic

Mechanistic metrics evaluate whether GRNs capture causal relationships from perturbation assays using the KnockTF database [64, 65]. This database identifies which transcription factors regulate specific genes by knocking out a TF and comparing gene expression levels before ($\text{exp}_{\text{before}}$) and after ($\text{exp}_{\text{after}}$) the knockout. Genes regulated by the TF will show changes in expression and can therefore be linked to that TF.

TF activity

For each TF knockout experiment targeting a transcription factor t , the $\log_2$ fold change is computed for all genes, yielding an observed response vector $\mathbf{y}_t$, where each entry y_{ti} reflects the expression change of gene g_i following the knockout:

$$y_{ti} = \log_2 \left(\frac{\text{exp}_{\text{after}}}{\text{exp}_{\text{before}}} \right) \begin{cases} > 0 & \text{upregulated} \\ < 0 & \text{downregulated} \\ = 0 & \text{no change} \end{cases} \quad (5.4)$$

The inferred GRN provides the predicted regulon of t : a set of target genes g_i each associated with an edge score w_{ti} reflecting how confidently the method predicts t to regulate g_i .

The TF activity score is estimated using decoupleR’s ULM method [66], which measures how correlated w_{ti} and y_{ti} are across all predicted targets of t . Since knocking out t removes its activating signal, we expect genes with a high w_{ti} to show the most negative y_{ti} , resulting in a significantly negative correlation score for a well-performing GRN. A true positive is therefore a TF present in the inferred GRN whose activity score is negative and statistically significant⁵ ($p_{\text{adj}} < 0.05$).

Forecasting

While TF activity directly correlates GRN edge scores with observed expression changes, forecasting simulates how a perturbation propagates through the network and checks whether the resulting predicted changes match the observed ones from KnockTF. The perturbation workflow is illustrated in Figure 5.6.

CellOracle [25] is used to fit a regression model on the expression matrix using the GRN as a connectivity skeleton. The expression of each target gene i can be expressed as a linear combination of the expressions of its regulators t :

$$\text{exp}(i) = \sum_t a_{it} \cdot \text{exp}(t) \quad (5.5)$$

This yields a coefficient matrix A , where only TF-TG links present in the GRN are kept and all other entries are set to zero.

The entire expression matrix $E \in \mathbb{R}^{\text{cells} \times \text{genes}}$ is collapsed into a pseudobulk mean expression vector $\mathbf{x} \in \mathbb{R}^{\text{genes}}$ containing average gene expression across cells. To simulate a TF knockdown, we initialise a perturbation vector $\Delta \mathbf{x}^{(0)} \in \mathbb{R}^{\text{genes}}$ where only the knocked-down TF_t is set to $-\mathbf{x}[t]$ and all other entries are zero, effectively removing its contribution from the network, and propagate this perturbation through A for three steps:

$$\Delta \mathbf{x}^{(s)} = A \cdot \Delta \mathbf{x}^{(s-1)}, \quad s = 1, 2, 3 \quad (5.6)$$

⁵The Benjamini–Hochberg procedure is used to control the false discovery rate (FDR): the expected proportion of false positives among all significant results. Setting $p_{\text{adj}} < 0.05$ ensures that at most 5% of declared true positives are expected to be false positives across all TF knockout experiments [67].

At $s = 1$, $\Delta \mathbf{x}^{(1)}$ is non-zero only for direct targets of TF_t , since $A \cdot \Delta \mathbf{x}^{(0)}$ picks out column t of A : the regression coefficients of all genes that TF_t directly regulates. At $s = 2$, $\Delta \mathbf{x}^{(2)}$ is non-zero for any gene that is regulated by a direct target of TF_t , as the non-zero entries of $\Delta \mathbf{x}^{(1)}$ now activate their own columns in A . At $s = 3$, the signal reaches one level deeper still.

The accumulated perturbation $\Delta \mathbf{x}^{(3)}$ thus captures both the immediate and downstream consequences of the knockdown up to three hops away in the network.

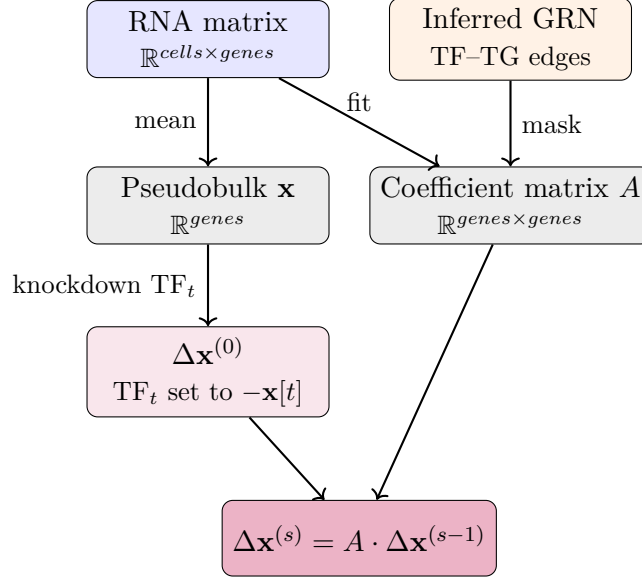


Figure 5.6: Perturbation propagation workflow ✱.

We then restrict both $\Delta \mathbf{x}^{(3)}$ and the observed KnockTF differential expression profile to their common genes, and compute the Spearman correlation ρ between them. A true positive is a TF whose predicted expression changes $\Delta \mathbf{x}^{(3)}$ are significantly positively correlated with the observed ones ($p_{\text{adj}} < 0.05$).

5.4.2 Predictive

Predictive metrics test whether the GRN's structure is consistent with the data it was inferred from, i.e. whether its wiring captures genuine statistical dependencies present in the data.

Omics

For each target in the GRN (gene or CRE), an XGBoost model is trained to predict that target's expression or accessibility from its assigned regulators (sources). The following relationships are modelled: Genes $\sim$ TFs, Genes $\sim$ CREs, and CREs $\sim$ TFs.

For example, when modelling the Genes $\sim$ TFs relationship, the expression of each target gene is predicted using only the transcription factors linked to it, with the GRN providing the inferred TF-TG interactions.

	Source	Target	Source mod.	Target mod.	Relationship
gtf	TF	Gene	RNA	RNA	TF $\rightarrow$ gene expression
gcre	CRE	Gene	ATAC	RNA	CRE $\rightarrow$ gene expression
cretf	TF	CRE	RNA	ATAC	TF $\rightarrow$ CRE accessibility

Table 5.1: Omics predictive metric variants

A true positive is a feature (e.g. a gene) included in the GRN as a target, whose molecular readout (e.g. expression) is significantly predicted from its GRN-assigned regulators ($p_{\text{adj}} < 0.05$).

Gene sets

This metric tests whether the TF regulons, sets of target genes g_i regulated by a TF, defined by the GRN are biologically meaningful, by checking if they align with known gene sets active in the RNA expression data. A gene set is a collection of genes known to share a specific biological function, such as a signalling response, a metabolic process, or a transcriptional programme [68].

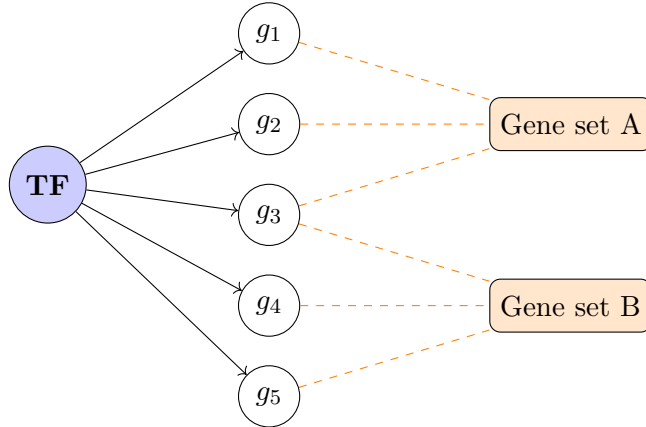


Figure 5.7: TF regulon and matched gene sets $\star$.

First, for each cell, a gene set activity score is computed by aggregating the expression of its member genes using decoupleR’s ULM method [66]. A gene set is considered active in the dataset if its score is positive in more than 20% of cells.

Then, for each TF in the GRN, a one-sided Fisher’s exact test⁶ checks whether any active gene set is overrepresented among its target genes relative to what would be expected by chance. A true positive is a gene set that is both active in the data and significantly enriched in at least one TF regulon ($p_{\text{adj}} < 0.01$).

⁶The one-sided Fisher’s exact test [69] assesses whether the overlap between two sets is larger than expected by chance, given their sizes relative to the total number of genes.

5.4.3 Literature based

The literature-based metrics test whether the GRN is consistent with existing biological knowledge using three complementary angles.

First, at the regulator level, by checking whether the TFs used as regulators in the GRN match TFs known to be active in the relevant cell types (marker TFs).

Second, at the co-regulation level, by testing whether TFs sharing many target genes in the GRN also physically interact (TF–TF pairs).

Third, at the edge level, by validating the different types of connections in the GRN: TF–TG edges validated with a curated reference database of known TF–TG interactions, TF–CRE edges against experimentally validated TF binding sites, CRE–gene edges against biologically supported regulatory links, and CREs themselves against independent evidence of functional activity.

In practice, this category splits into two groups: Literature metrics (grey), comprising TF pairs, TF markers, and TF–TG links; and Genomic metrics (green), comprising TF binding, CRE–TG links and functional CREs.

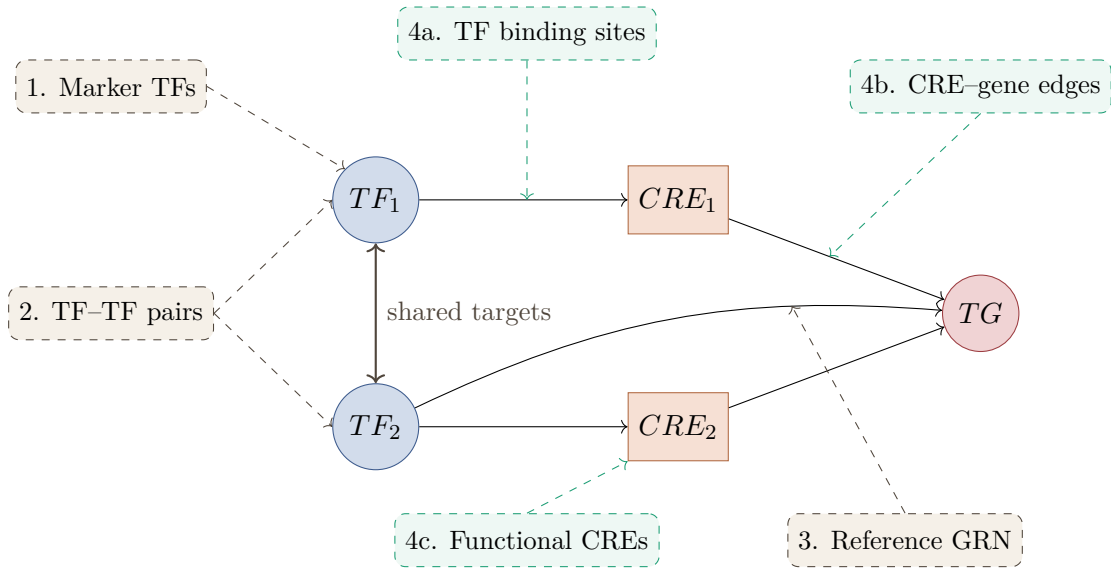


Figure 5.8: Literature-based (grey) and genomic (green) validation strategies for GRN evaluation ✱.

6. Benchmark

6.1 LINGER triplets extraction strategy

When constructing the final GRN for GRETA, we opted to build TF–CRE–TG triplets from all LINGER outputs, assuming that leveraging the full set of available information would be beneficial. An alternative approach would have been to rely solely on the final γ (TF–TG) interactions, without constructing triplets. To validate our triplet-based strategy, we compared both approaches by computing their $\Delta_{F_{0.1}}$ score (Figure 6.1).

CRE-specific metrics can now be evaluated, all gene set metrics show improvement, and mechanistic and literature-based metrics remain largely unchanged. We see a decrease in performance for the Omics Gene $\sim$ TFs metric.

The improvement on Gene Sets tasks may reflect the selectivity of the triple-product score: an edge receives a high score only when all three signals converge, meaning that the TF binds an accessible CRE (α_{tk}), and both CRE and TF have strong Shapley values for the target gene’s expression (β_{ki} and γ_{ti}). This may make TF regulons more biologically coherent, as their target genes are all connected to the TF through CRE-mediated regulatory paths.

The decrease in Omics Gene $\sim$ TFs is likely the direct counterpart of this specificity gain. The γ matrix directly reflects predictive TF–TG relationships, as it is the output of LINGER’s networks trained to predict gene expression from TF inputs. In the triplet GRN, a TF–TG link is only retained when also supported by CRE evidence, adding a constraint that may exclude otherwise predictive TF–TG pairs.

The absence of any decrease on the Reference GRN metric, which directly evaluates TF–TG edge overlap against CollecTRI¹, suggests this is specific to the expression-prediction objective² rather than a general degradation of TF–TG links.

¹curated resource of TF–target gene interactions [28].

²The Omics Gene $\sim$ TFs fits an XGBoost model to predict each target gene’s expression from its assigned TF regulators only (Table 5.1).



Figure 6.1: Evaluation of the triplet-based GRN construction strategy from LINGER outputs. The figure reports the change in $F_{0.1}$ score ($\Delta F_{0.1}$) relative to its baseline approach that uses only TF–TG interactions from the final γ matrix $\star$.

6.2 LINGER baseline

We compared LINGER against its baseline, which constructs the GRN directly from bulk prior matrices and uses single-cell data only as a feature mask, without neural network training. Both approaches operate on the same genes, TFs, and peaks, but differ in how edge scores are assigned: the baseline relies on bulk prior values, whereas LINGER derives data-driven scores by training per-gene neural networks on single-cell data.

Mechanistic and literature-based metrics showed negligible improvement across all tasks ($\max(\Delta F_{0.1}) < 0.03$).

Predictive metrics yielded contrasting results: modest gains were observed for the Omics task, while the Gene Sets task showed a substantial improvement ($\min(\Delta F_{0.1}) > 0.2$) for all databases. This is consistent with how the metric is defined: gene set activity is computed directly from the PBMC expression data, so scores derived from networks trained on PBMC single-cell data could better reflect which TF–gene relationships are active in this context than bulk-derived scores.

Genomic metrics were more mixed: only four databases within the CRE task showed improvement, while all remaining categories either declined or remained unchanged.

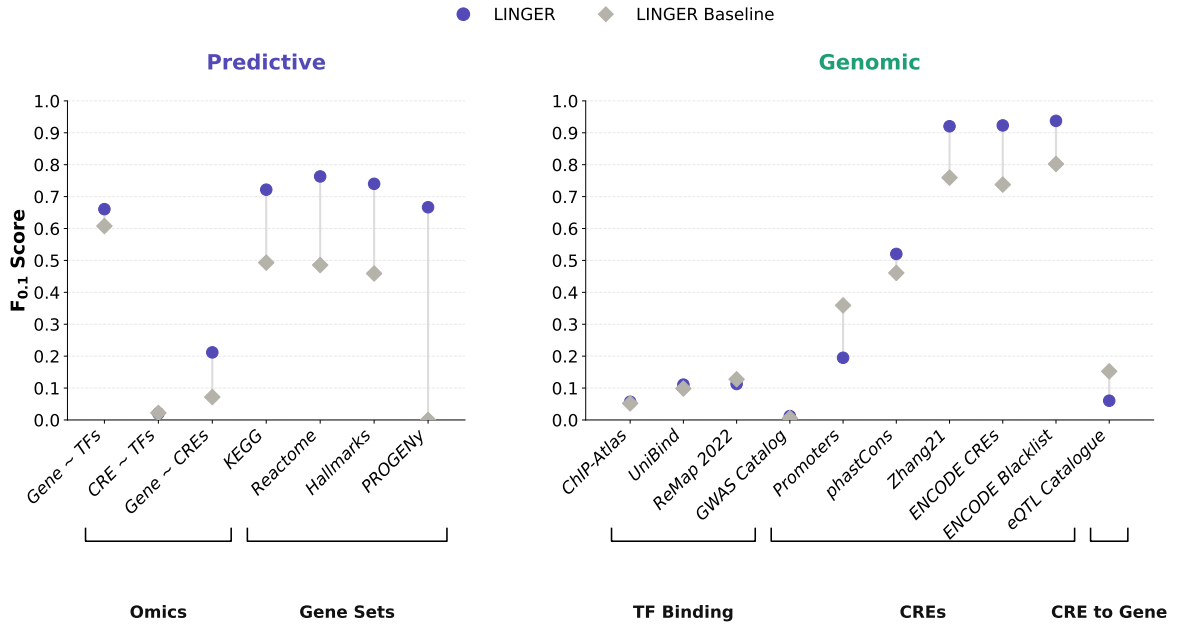


Figure 6.2: Baseline comparison of LINGER and its baseline across Predictive and Genomic tasks ✱.

6.3 Similarity

To quantify the agreement between GRN inference methods, we used the overlap coefficient, defined in GRETA as the intersection size between two sets divided by the smaller set’s size. This metric accounts for differences in network size, ensuring that comparisons between networks of vastly different scales remain meaningful.

We compared the GRNs obtained from the same PBMC dataset across all methods. Most methods inferred GRNs with similar numbers of TFs, edges, and target genes per TF (median TFs = 202, edges = 4,101, genes = 1,746, regulon size = 28.5) (Figure 6.3). Dictys, LINGER and its baseline, and SCENIC+ produced considerably larger networks, with edge counts exceeding 18k and regulon sizes ranging from 57 to 200 target genes per TF. DeepMAPS stood apart at the other extreme, yielding the smallest network with only 13 TFs, 149 edges, and a regulon size of 11.5, suggesting it captures only a narrow set of regulatory relationships in this dataset.

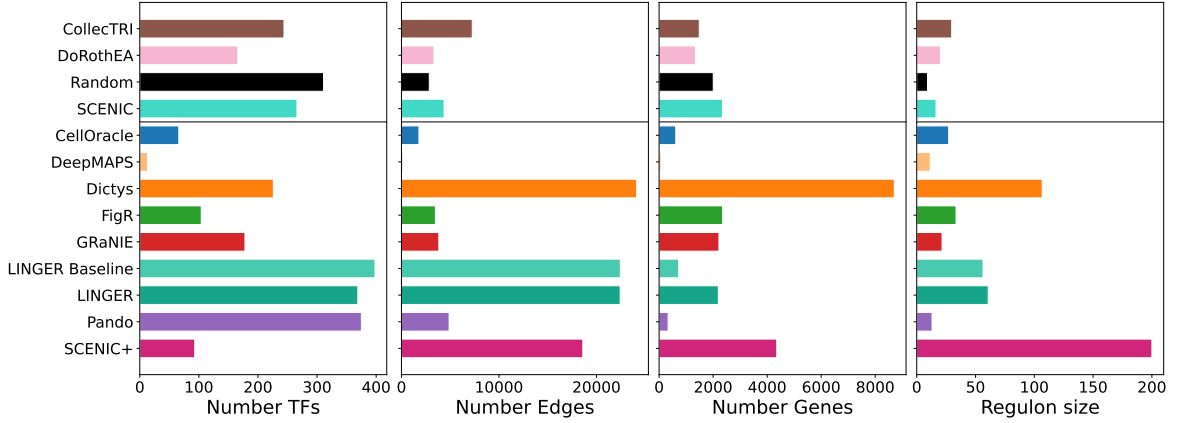


Figure 6.3: Network size statistics across GRN inference methods. Transcription factors TF, Regulon size is the average number of target genes per TF.

Pairwise comparisons of network nodes revealed moderate agreement at the TF and target gene levels across methods (mean overlap coefficient TFs = 0.63; mean overlap coefficient genes = 0.47) (Figure 6.4).

The strongest TF concordance was observed between LINGER and its baseline, as both operate on the same feature mask and therefore consider the same set of expressed TFs. The low overlap at the TG and CRE levels reflects that different scoring leads GRETA to select largely different top edges.

SCENIC+ and Pando also showed notable TF agreement (overlap coefficient = 0.86), which can be explained by their shared use of the JASPAR [37] and CIS-BP [70] motif databases for TF binding site prediction: Pando relies on both directly, while SCENIC+ incorporates them as part of its broader curated collection of over 34,000 motifs drawn from 29 different databases.

At the CRE level, overlap was substantially lower and more variable (mean overlap coefficient CREs = 0.29), with Dictys standing out as the method most consistently sharing regulatory regions with others (mean overlap coefficient CREs = 0.48), while methods such as Pando showed near-zero CRE overlap with most approaches.

Similarly, at the gene level, Dictys exhibited the broadest agreement with other methods (mean overlap coefficient genes = 0.73), reflecting its tendency to produce large networks covering a wide range of target genes.

Edge overlap remained minimal across all method pairs (mean overlap coefficient edges = 0.06) (Figure 6.5), with only the two literature-curated networks CollecTRI and DoRothEA (overlap coefficient = 0.34), and SCENIC+ and Pando (overlap coefficient = 0.11), standing out as exceptions, though their overlap remains low.

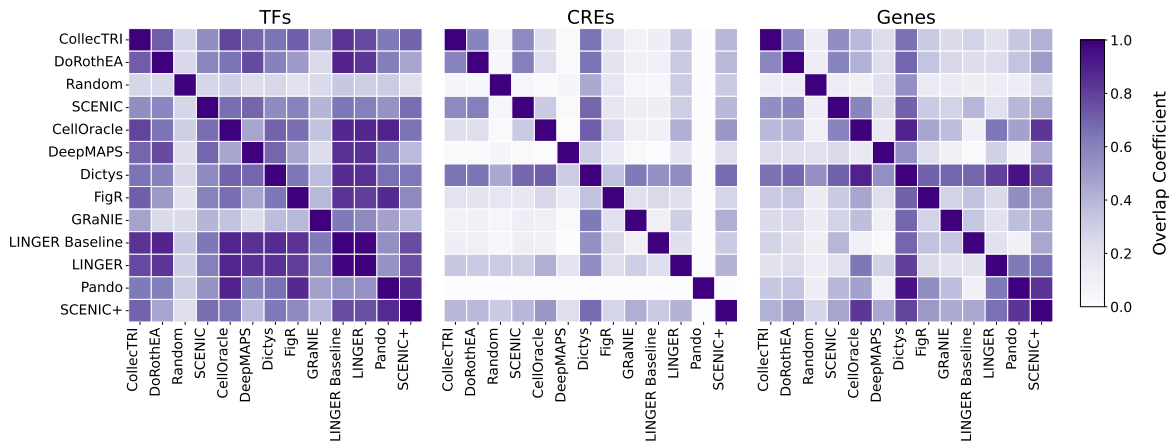


Figure 6.4: Pairwise overlap coefficients between GRNs at the TF, CRE, and target gene levels

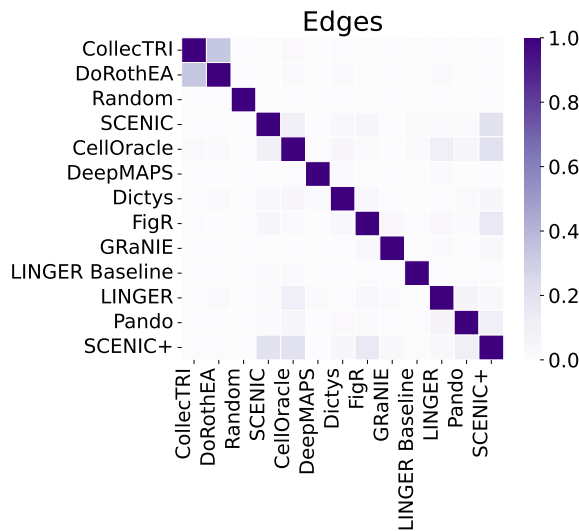


Figure 6.5: Pairwise overlap coefficients between GRNs at the edge level

6.4 Stability

Stability in GRN inference reflects both reproducibility across runs and robustness to reduced data coverage. We assessed both for LINGER, using the pituitary gland dataset [71] to remain aligned with GRETA’s experimental setup and enable direct comparison with their results (Appendix A.5).

DeepMAPS was excluded from this analysis due to its poor overall performance in benchmarking and the difficulties encountered in reproducing its original implementation.

6.4.1 Reproducibility across runs

To assess reproducibility, LINGER was run five times on the pituitary gland dataset, and the resulting networks were compared pairwise (10 comparisons in total) at the TF, CRE, gene, and edge levels (Figure 6.6).

LINGER recovered the same TFs almost perfectly across runs (mean overlap coefficient = 0.96). Gene recovery was slightly less consistent (mean overlap coefficient = 0.82), while CRE reproducibility was moderate (mean overlap coefficient = 0.68). Edge reproducibility was the most affected, with a mean overlap of 0.36.

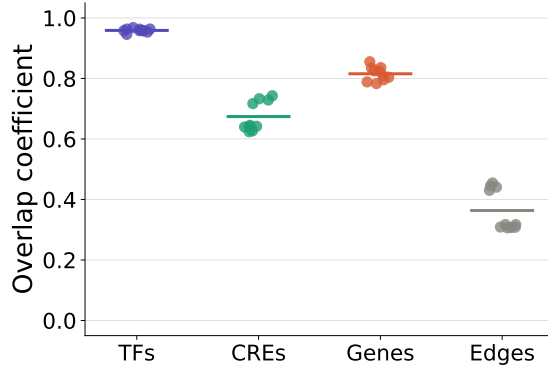


Figure 6.6: Pairwise overlap coefficients between five LINGER runs on the full pituitary gland dataset at the TF, CRE, gene, and edge levels ✱.

6.4.2 Robustness to data downsampling

To assess sensitivity to data coverage, we progressively downsampled the pituitary gland dataset along two axes: the number of cells (with features held fixed) and the number of features (with cells held fixed), and compared the resulting networks to the one inferred from the full dataset using the overlap coefficient at the TF, gene, and edge levels.

The stability score is defined in GRETA as the area under the overlap coefficient curve (AUC) across downsampling levels, with values close to one indicating robustness and values close to zero indicating high sensitivity. Five random seeds were used at each level.

When cells were downsampled with features held fixed, LINGER produced networks at all levels, with TF recovery remaining consistently high across cell counts (Figure 6.7, left). When features were downsampled with cells held fixed, LINGER failed to produce any network at the two lowest feature counts (5,120 and 10,240), suggesting a minimum feature coverage requirement. Above this threshold, recovery improved and TF overlap reached near-perfect values (Figure 6.7, right).

The stability scores confirm that LINGER is more robust to cell downsampling than to feature downsampling at the TF, gene, and edge levels (Figure 6.8): TF stability reaches 0.90 when cells are reduced but only 0.62 when features are reduced, identifying feature coverage as the primary bottleneck for stable inference. This pattern holds across all levels, with gene stability following the same trend and edge stability remaining low under both axes (AUC = 0.216 and 0.161).

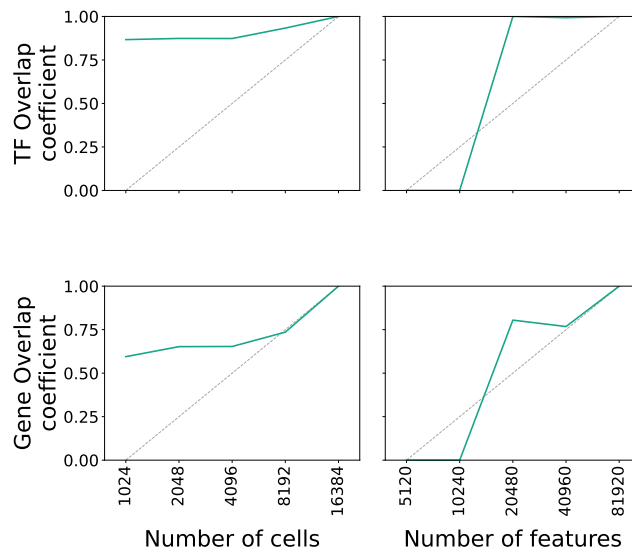


Figure 6.7: Overlap coefficients of TFs (top) and target genes (bottom) between GRNs inferred from downsampled pituitary gland dataset and the one inferred using all available data ✱.

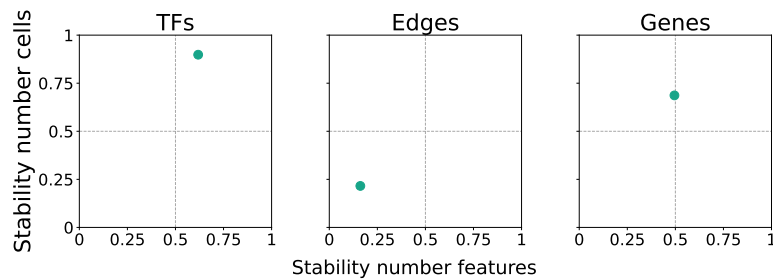


Figure 6.8: Stability scores (AUC) of LINGER computed under cell downsampling (y-axis) and feature downsampling (x-axis) at the TF, edge, and gene levels ✱.

6.5 GRN evaluation on the PBMC dataset

6.5.1 Choice of the dataset

The benchmark was performed on the PBMC dataset (Appendix A.2). Following the dataset processing pipeline described in Section 5.2, 9,631 cells were retained and used as input to the GRN inference methods.

This choice is motivated by two considerations: ground-truth coverage and widespread adoption. PBMCs are well-studied cell populations in regulatory genomics, with broad representation across public databases, providing ground-truth entries across all evaluation databases used in this benchmark. This breadth matters given the precision-weighted $F_{0.1}$ scoring used throughout: regulatory interactions that are true but absent from a sparse ground truth are penalised as false positives, deflating scores. A well-populated ground truth reduces this effect and makes evaluation more reliable.

Beyond coverage, this dataset has been widely adopted as a reference for GRN benchmarking. It was used as a primary evaluation set in SCENIC+ [30], LINGER [35], and DeepMAPS [4], as well as in the original GRETA paper [5].

6.5.2 Global performance overview

Figure 6.9 summarises the performance of 12 GRN inference methods across the three metric categories in GRETA on the PBMC dataset. Scores reported in this section are mean $F_{0.1}$ scores averaged across all databases within the relevant task.

Mechanistic metrics reveal a near-universal failure to capture causal perturbation effects: TF activity and Forecasting scores are close to zero for most methods, confirming that current GRNs do not reliably propagate TF knockout signals through their network structure. SCENIC+ performs best (0.143), followed by Dictys (0.050), while all other multimodal methods remain below 0.025.

Predictive metrics paint a more nuanced picture. The Omics tasks show a clear hierarchy: SCENIC+ leads (0.337), followed by Dictys (0.301) and LINGER (0.298), while DeepMAPS is a clear outlier at 0.027. Gene Sets tasks yield substantially higher scores and discriminate methods well: CellOracle achieves the highest score (0.796), followed by LINGER (0.723) and SCENIC+ (0.630).

Literature-based metrics reveal a mixed picture. Scores are uniformly low across all methods (below 0.026), with CellOracle and Pando narrowly leading, suggesting that inferred GRNs recover only a small fraction of curated interactions. Notably, CollecTRI and DoRothEA are excluded from this task since CollecTRI is itself the ground truth.

For TF Markers, multimodal methods outperform prior-knowledge baselines: CellOracle (0.417) and FigR (0.415) lead ahead of CollecTRI (0.275) and DoRothEA (0.289), suggesting that TF marker recovery is not simply a function of curated knowledge. For TF Pairs, Dictys leads clearly (0.389), followed by DoRothEA (0.323) and Pando (0.315). For the reference GRN task, all methods perform poorly.

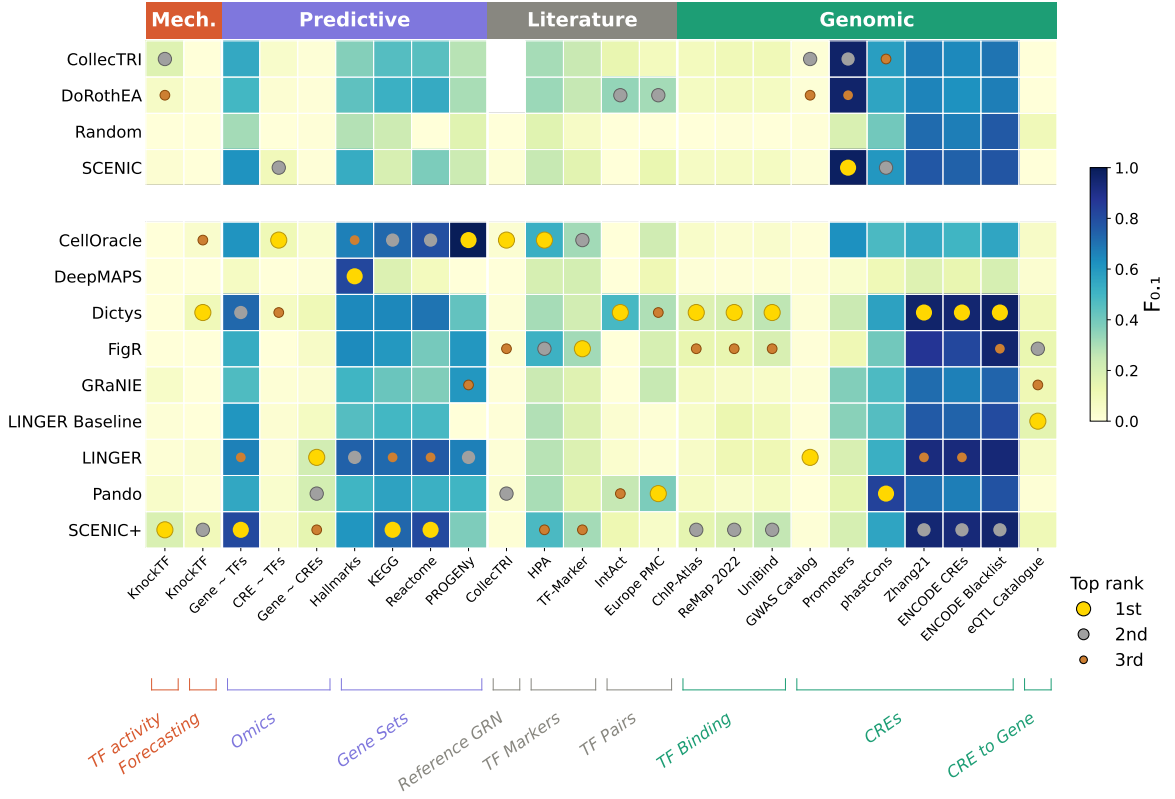


Figure 6.9: $F_{0.1}$ scores across 10 evaluation tasks grouped by metric category for 12 GRN inference methods. Gold, silver and bronze circles indicate the top three methods per task. Unimodal methods are shown in the top panel and multimodal methods in the bottom panel $\star$.

Genomic metrics cover TF Binding, CREs, and CRE to Gene tasks. For TF Binding, scores are low overall but not uniform: Dictys (0.214) and SCENIC+ (0.199) lead, while Random remains near zero (0.003), confirming that TF binding recovery reflects genuine inference rather than background signal.

CRE-level metrics, by contrast, are high across nearly all methods, including Random (0.458). This is expected: since CRE databases annotate functional regions genome-wide, any method that links genes to nearby accessible regions will passively overlap a large fraction of them.

The CRE to Gene task yields more discriminative scores, though performance remains poor overall, with LINGER Baseline (0.152) and FigR (0.133) leading.

DeepMAPS underperforms consistently across almost all categories, with one notable exception: it achieves the highest score among all methods on the Gene Sets task, specifically on the Hallmarks database, with an $F_{0.1}$ of 0.815.

One possible explanation involves the construction of Hallmarks [72]: compared to other gene set databases that are derived from curated molecular interactions or perturbation experiments, Hallmarks groups genes based on coherent co-expression patterns observed across many datasets, which may overlap naturally with the co-expressed gene modules that DeepMAPS produces (Section 4.4.2). PROGENy [73],

for instance, defines gene sets from perturbation experiments by identifying genes whose expression changes when a specific upstream regulator is activated or inhibited, a more causal criterion that a co-expression-based method would not be expected to satisfy.

Figure 6.10 shows that Dictys achieves the most gold medals (8), indicating the strongest peak performance across tasks. SCENIC+ follows with the highest overall number of medals, reflecting consistent high performance. CellOracle, LINGER and FigR maintain strong overall performance, with repeated appearances among the top-ranked methods, while most other methods receive fewer or no medals.

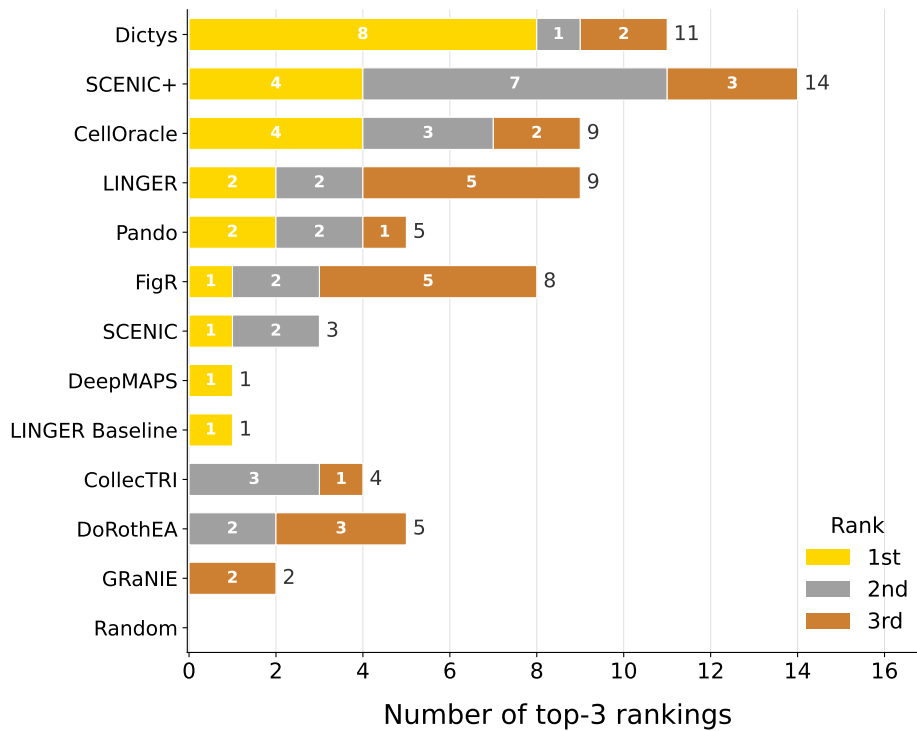


Figure 6.10: Number of gold, silver, and bronze medals obtained by each method across all evaluation tasks ✱.

6.5.3 Best methods per category

While the previous section examined each task individually, this section shows method performance by averaging $F_{0.1}$ scores across all tasks within each metric category, summarised in Figure 6.11.

Mechanistic metrics are the most challenging, with a very low median score (0.020) and SCENIC+ leading (0.143), followed by CollecTRI and Dictys, confirming the difficulty of recovering causal regulatory structure.

Predictive metrics yield substantially higher scores (median 0.343), where CellOracle, LINGER and SCENIC+ perform best, while Random still reaches non-trivial performance.

Literature-based metrics show moderate results (median 0.193), with DoRothEA and Dictys leading.

Finally, Genomic metrics show the highest overall scores (median 0.370) but limited separation between methods, with Dictys and SCENIC+ leading; even Random performs relatively well, indicating that preserving structural constraints such as CRE–gene windowing and degree distributions (Section 5.3.4) is sufficient to recover a substantial fraction of genomic overlaps even in the absence of true biological signal.

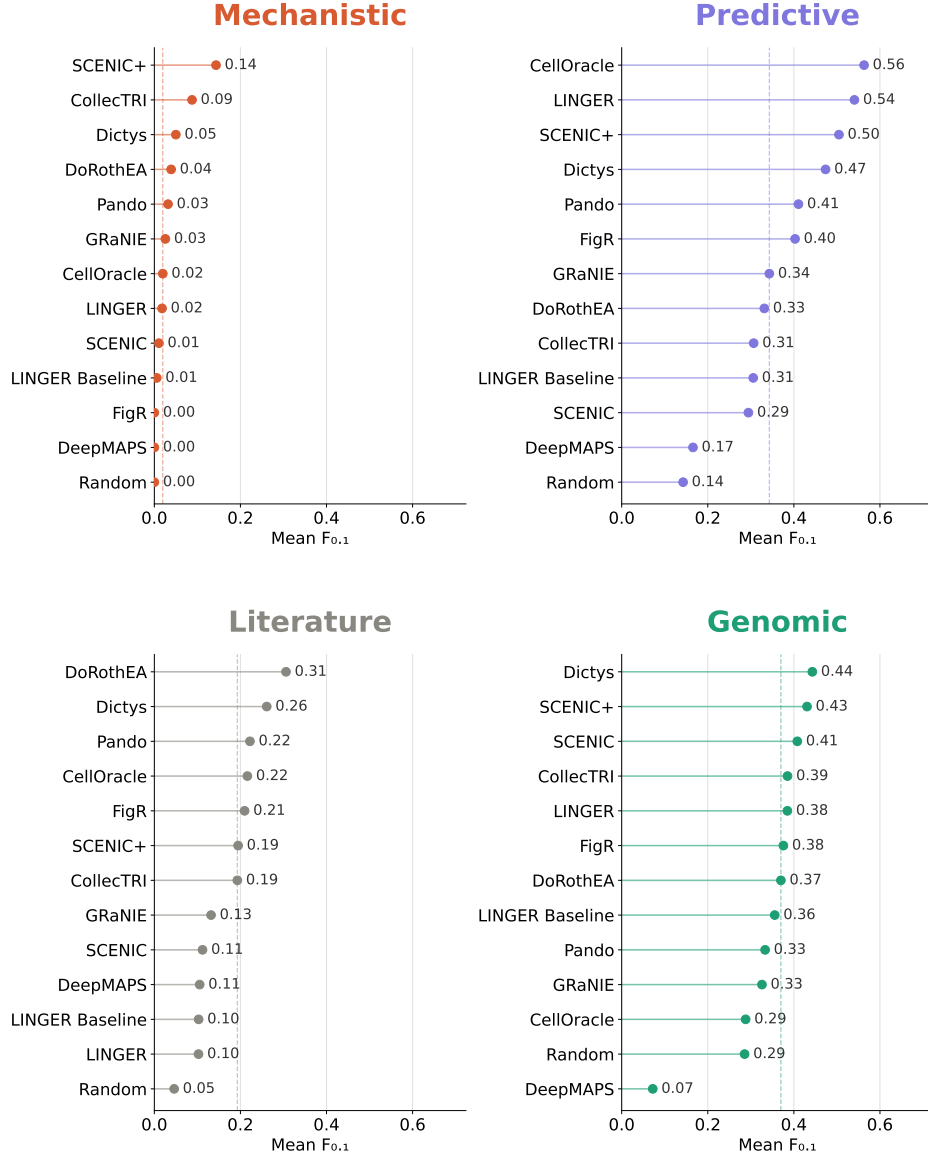


Figure 6.11: Mean $F_{0.1}$ scores per GRN inference method, grouped by metric category. Within each panel, methods are sorted by their mean score averaged across all tasks in that category. The dashed vertical line indicates the median score across methods *

6.6 TF activity

We benchmarked LINGER’s TF activity module described in Section 3.7 against SCENIC+, the state-of-the-art method for inferring GRNs. Both methods estimate TF activity at single-cell resolution, enabling the identification of cell-type-specific transcriptional regulators.

The output of SCENIC+ is a set of enhancer-driven regulons (eRegulons), each defined by a single TF together with its set of target regions (genomic peaks enriched for that TF’s binding motif) and its set of target genes linked to those regions, forming TF–CRE–TG triplets.

Once eRegulons are constructed, TF activity is quantified per cell using the AUCCell algorithm [56]. For each cell j and each eRegulon e , all I genes in the RNA expression matrix are ranked by their expression level in cell j , from most to least expressed.

A recovery curve is then built by walking down this ranking from rank 1 to rank I : at each position, the cumulative number of eRegulon e ’s target genes recovered so far is recorded, such that the curve rises by one step each time a target gene is encountered and remains flat otherwise.

A TF whose target genes are highly expressed will therefore produce a curve that rises steeply at the top of the ranking, whereas an inactive TF will produce a flat, slowly rising curve. The gene-based AUC score is defined as the area under this recovery curve restricted to the top 5% of the ranking. A high AUC indicates that the target genes of eRegulon e are concentrated at the top of the expression ranking in cell j , reflecting high activity of the associated TF.

This produces a cell-eRegulons matrix in which each entry reflects how strongly a given TF’s target gene set is active in a given cell, analogous to the LINGER TF activity matrix. However, the two methods differ in how they arrive at this score.

LINGER measures the excess co-expression of a TF’s target genes above a null expectation, using the RNA expression matrix E and the trans-regulatory strength γ . SCENIC+ is entirely rank-based: it does not use TF expression at all but relies on the position of TF-assigned target genes in the per-cell expression ranking.

Once single-cell TF activity scores are obtained from each method, scores are averaged within each cell type, yielding one score per TF per cell type. To make scores comparable across TFs with different dynamic ranges, each TF’s activity is z-scored across cell types. Since the PBMC dataset contains 19 different cell types (Appendix A.2), we restricted the analysis to cell types with at least 300 cells, retaining 9 cell types for downstream comparison.

Among the 407 TFs identified by LINGER, 54 overlapped with the 66 TFs identified by SCENIC+. Figure 6.12 shows the heatmaps of these 54 shared TFs across the 9 cell types. Both display broadly similar patterns, suggesting that the TF activity estimates produced by LINGER are consistent with those of SCENIC+.

Table 6.1 reports, for each cell type, the TFs that appear in the top 5 of both methods³. The overlap is non-empty in 7 out of 9 cell types, with a consistent recovery of 3 shared TFs in most cases.

Cell type	Shared top TFs
CD4 Naive	FOXO1 , LEF1 , TCF7
CD8 Naive	FOXO1 , LEF1 , TCF7
CD4 TCM	ETS1 , MAF , ZEB1
CD16 Mono	IRF2 , LMO2 , MAFB
NK	TBX21 , RUNX3 , MYBL1 , EOMES
CD8 TEM_2	TBX21 , RUNX3 , MYBL1 , EOMES
Intermediate B	PAX5 , SPIB , TCF4
CD14 Mono	—
CD8 TEM_1	—

Table 6.1: Overlap in top TF activity rankings between LINGER and SCENIC+

The identified transcription factors are consistent with established cell-type-specific regulatory programmes. For example, *FOXO1* and *TCF7* are canonical regulators of naive T cell⁴ identity, shared here by both the CD4 and CD8 Naive subsets [75, 76].

Similarly, *RUNX3* is a key regulator of cytotoxic T lymphocyte differentiation [77], consistent with its activity in both the NK and CD8 TEM_2 clusters. *PAX5* is a well-established marker of B cell identity [78].

The two exceptions are CD14 Mono and CD8 TEM_1, where no overlap was found between the top 5 TFs of the two methods, suggesting greater divergence in the regulatory signals captured for these cell types.

Spearman correlations between LINGER and SCENIC+ TF activity z-scores across the 54 shared TFs are reported in Appendix A.6, and are statistically significant in 8 out of 9 cell types, confirming the broad concordance observed in the heatmaps.

³For each cell type, the five TFs with the highest activity score were selected independently from each method, and the two sets were then intersected.

⁴A T cell, or T lymphocyte, is a type of white blood cell that plays a central role in cell-mediated immunity. There are several subsets of T-cells, such as naive T-cells (CD4 Naive, CD8 Naive), helper T cells (CD4 TCM), cytotoxic T cells (CD8 TEM_2) and a broader category of cytotoxic lymphocytes (CD8 TEM_2, NK) [74, 75].

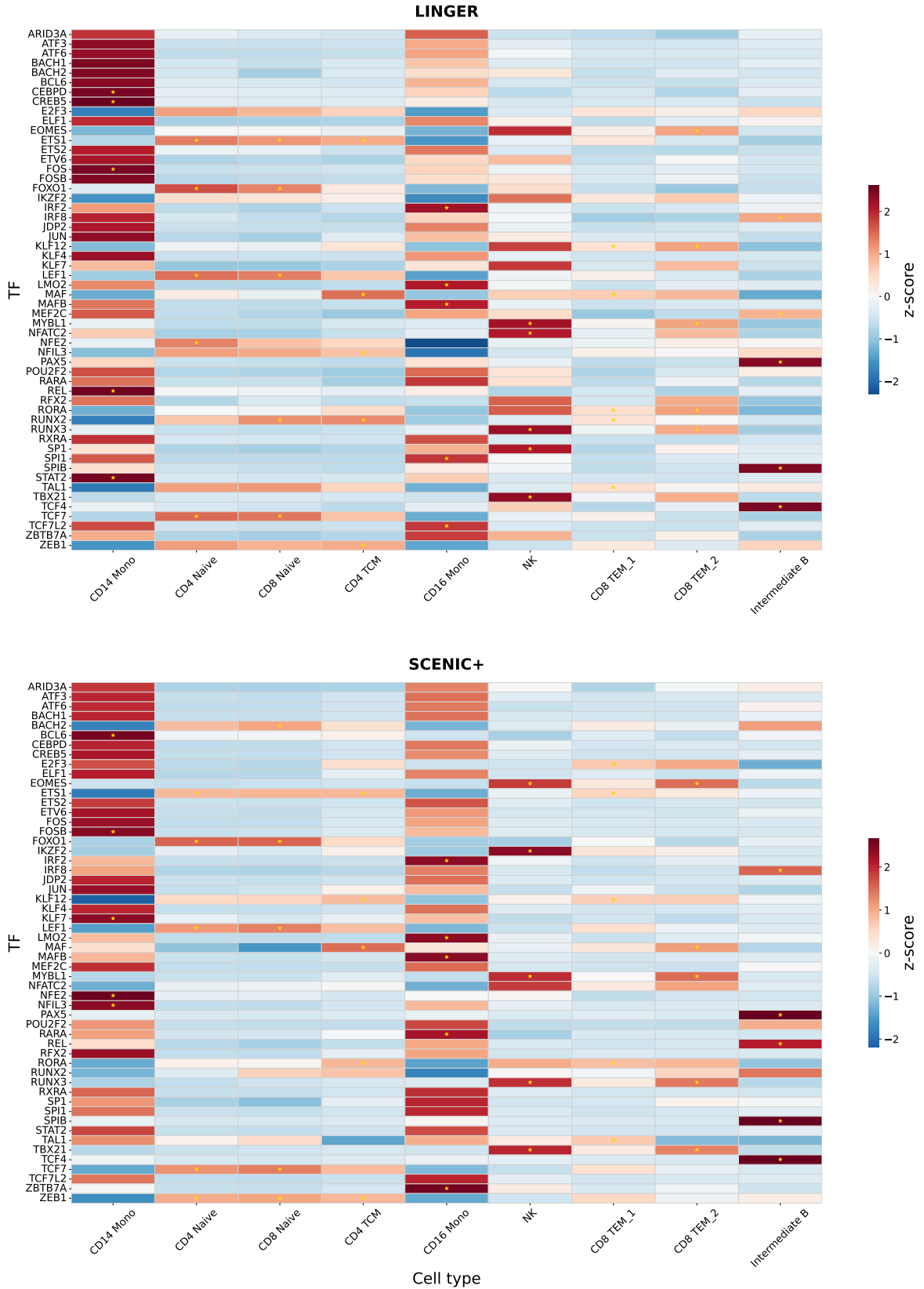


Figure 6.12: Heatmaps of TF activity z-scores for the 54 TFs shared between LINGER and SCENIC+ across 9 cell types. Gold stars (★) indicate TFs ranking in the top 5 for a given cell type ✱.

7. Conclusion

7.1 Summary

This thesis integrated two deep learning-based GRN inference methods, LINGER and DeepMAPS, into the GRETA benchmarking framework, and benchmarked them alongside existing approaches on the PBMC dataset. LINGER’s chromosome-level processing was additionally parallelised, yielding substantial speedups across preprocessing, training, and GRN reconstruction.

The benchmark results suggest that deep learning does not systematically outperform existing methods. LINGER remains competitive, particularly on predictive tasks, and its TF activity estimates show broad concordance with those of SCENIC+. DeepMAPS underperforms across most categories, which may partly reflect the two adaptations required to reproduce the original pipeline.

More fundamentally, while LINGER uses deep learning to directly model regulatory causality, DeepMAPS employs it to capture statistical co-activity patterns between genes and cells, with regulon construction handled by separate motif-matching steps. This distinction suggests that LINGER makes more effective use of deep learning for GRN inference.

7.2 Limitations

Incomplete and non-cell-type-specific ground truth. The most fundamental limitation of any GRN benchmark is the absence of a complete ground truth. No exhaustive, experimentally validated map of TF–TG regulatory interactions exists, and the databases used in GRETA cover only a fraction of true regulatory relationships.

Two compounding factors motivate evaluating all methods at the cell-population level rather than at cell-type resolution. First, ground-truth databases annotating which interactions are active in specific cell types remain rare. Second, not all methods integrated in GRETA natively produce cell-type-specific GRNs, making a cell-type-level comparison across methods impractical.

Together, these constraints mean that the cell-type-specific GRNs that LINGER and DeepMAPS are specifically designed to produce cannot currently be evaluated in a systematic and fair way.

Training–evaluation data overlap. Methods that incorporate large external datasets as prior knowledge risk being evaluated on data derived from the same source as their training set. LINGER provides a concrete example: its per-gene neural networks are pre-trained on 201 paired bulk samples from the ENCODE project (Section 3.2), while GRETA’s `encode` database (cis-regulatory element annotations derived from the same ENCODE experiments) serves as one of its evaluation targets. Performance on this metric may therefore partly reflect regulatory patterns encountered during pre-training rather than genuine inference from single-cell data.

More broadly, this issue highlights a general challenge for benchmarking methods that rely on large prior knowledge bases: as the same public resources tend to be reused across training and evaluation, disentangling what a method has learned from what it has memorised becomes increasingly difficult.

Single dataset. The benchmark was performed on a single dataset. While the PBMC dataset was chosen for its broad ground-truth coverage across all GRETA evaluation databases and its widespread adoption as a reference for GRN benchmarking [30, 35, 4, 5], results obtained in this specific biological context may not generalise to other tissues or disease settings.

Evaluating methods across a wider range of datasets and cell types would strengthen the conclusions drawn here, but was precluded by the substantial computational cost of running all methods on all the different datasets available in GRETA.

A. Appendix

A.1 Categories of GRN inference methods

Method		Prog. 	Cell type 	Metacell 	Gene Regulatory Network			Inputs		
					 TF-CRE	 CRE-Gene	 TF-Gene	RNA	ATAC	Paired
Correlation	FigR		✗	✗	Motif enrichment	Spearman	Spearman	✗	✗	✓
	GRaNI		✗	✗	Motif enrichment	Pearson	N/A	✗	✗	✓
Regression	CellOracle		✓	✗	Motif enrichment	Co-accessibility	Ridge regression	✓	✗	✓
	Pando		✗	✓	Motif enrichment	N/A	Linear regression	✗	✗	✓
	SCENIC+		✓	✓	Motif enrichment	Gradient boosting	Gradient boosting	✓	✓	✓
	SCENIC		✓	✗	Motif enrichment (promoter)	N/A	Gradient boosting	✓	✗	✗
D.S.	Dictys		✓	✓	Motif enrichment + footprint	N/A	Stoch. diff. eq.	✗	✓	✓
D.L.	DeepMAPS		✓	✗	Motif enrichment	Graph autoencoder	Regulon construction	✗	✗	✓
	LINGER		✓	✓	Motif enrichment	Neural net (SHAP)	Neural net (SHAP)	✗	✗	✓

Figure A.1: **Summary of current GRN inference methods for paired multi-omic data included in this paper** Prog., the program which the method was implemented in; Cell type, whether the method produces cell type-specific GRNs; Metacell, whether the method aggregates single cells into metacells (averaged expression profile of multiple similar cells); TF-CRE, the main method used to infer TF-CRE links; CRE-Gene, the main method used to infer CRE-gene links; TF-gene, the main method used to infer TF-gene links. Inputs: RNA, whether the method is compatible with scRNA-seq data alone; ATAC, whether the method is compatible with scATAC-seq data alone. All methods can take paired RNA and ATAC data as input; Prob., probabilistic model; D.S, dynamical systems; D.L, deep learning; diff. eq., differential equation. N/A cells indicate that the respective step for identifying TF-CRE, CRE-Gene, or TF-Gene links is not done for a given method. Adapted from [3] .

A.2 PBMC dataset

The PBMC 10k dataset [43] jointly profiles gene expression and chromatin accessibility from approximately 11,898 cryopreserved peripheral blood mononuclear cells of a healthy 25-year-old female donor. Following GRETA’s processing pipeline (Section 5.2), 9,631 cells were retained across 19 cell types, measured across two modalities: single-cell RNA-seq (16,384 genes) and single-cell ATAC-seq (65,536 peaks). Cell types with fewer than 300 cells were excluded for TF activity analysis (Section 6.6) but retained for all other analyses.

PBMC10k — cell type distribution

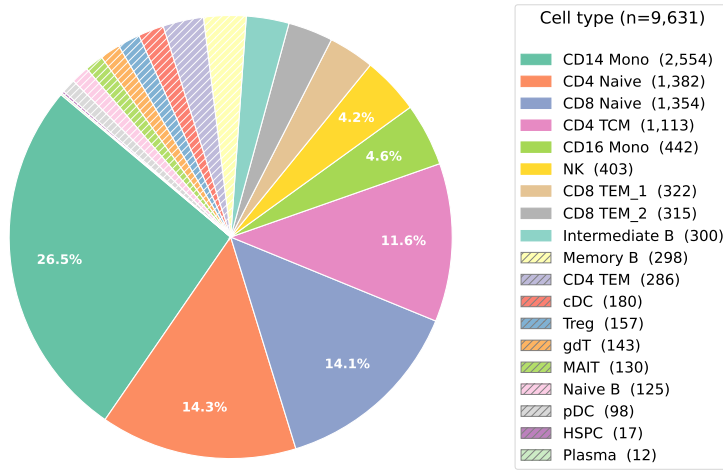


Figure A.2: PBMC dataset cell type distribution ✱.

A.3 GRETA Evaluation metrics and processed databases

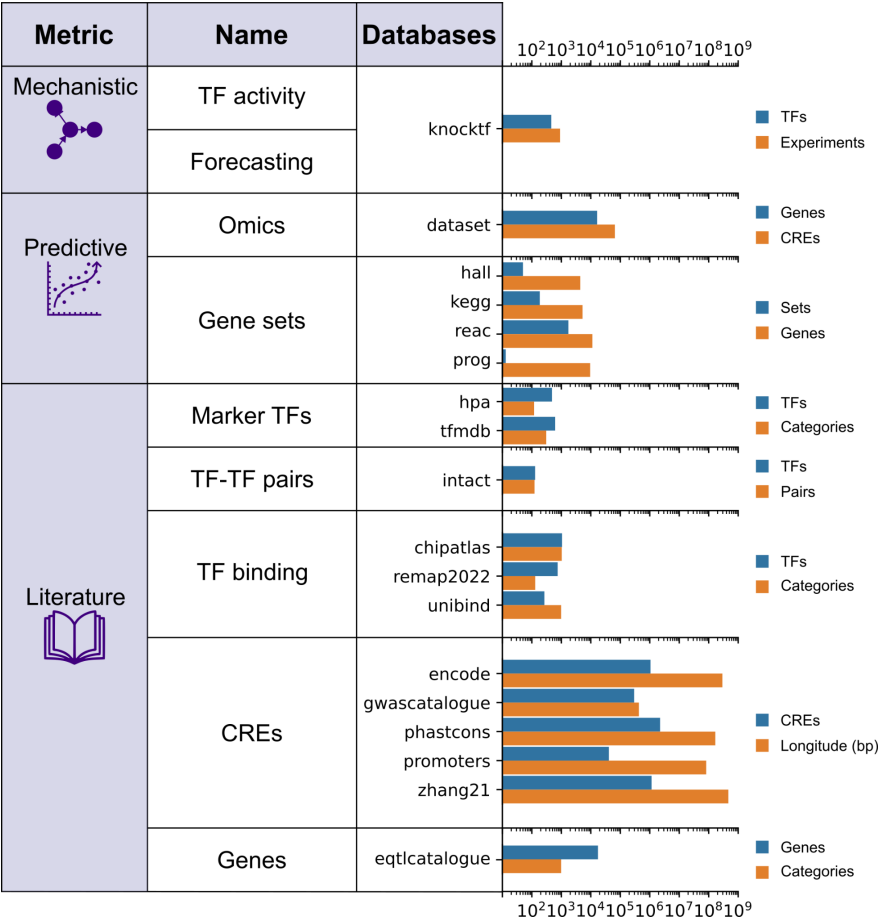


Figure A.3: **Evaluation metrics and processed databases.** List of evaluation metrics, the databases employed to calculate evaluation scores and their statistics such as number of elements, or number of cell-type or tissue labels. The Omics does not use a database as it used the original single-cell data. From [5].

A.4 GRETA Definition of evaluation metrics

Type	Name	Objective	Definition
Mechanistic	TF activity	Identify altered TFs in context-specific perturbation experiments	- TP = TF included in GRN with significant activity score for a perturbation experiment - FP = TF included in GRN but with insignificant activity score for a perturbation experiment - FN = TF not in GRN but in perturbation experiment
	Perturbation	Anticipate changes in gene expression after a context-specific TF perturbation simulation propagated through the GRN	- TP = TF targets included in GRN with significant Spearman correlation between predicted and observed changes in gene expression - FP = TF targets included in GRN but with insignificant Spearman correlation between predicted and observed changes in gene expression - FN = TF not in GRN but in perturbation experiment
Predictive	Omics	Independently predict omic feature readouts from other omic features from the original omics dataset. For example $\text{Gene} \sim \text{TF}_1 + \dots + \text{TF}_n$. The features to use are based on the inferred GRN topology	- TP = omic feature included in GRN with significant Spearman correlation between predicted and observed molecular readouts - FP = omic feature included in GRN but with insignificant Spearman correlation between predicted and observed molecular readouts - FN = omic feature not in GRN but in omic dataset
	Gene sets	Inclusion of gene sets in GRN that are predicted to be present in the original omics dataset	- TP = gene set in GRN and omics data - FP = gene set in GRN but not in omics data - FN = gene set not in GRN but in omics data
Literature	Marker TFs	Presence of context-specific TF markers in GRN	- TP = TF in GRN and marker collection - FP = TF in GRN but not in marker collection - FN = TF not in GRN but in marker collection
	TF-TF pairs	Presence of known TF-TF interactions in GRN	- TP = TF-TF in GRN and in interactions collection - FP = TF-TF in GRN but not in interactions collection - FN = TF-TF not in GRN but in interactions collection
	TF binding	Presence of TF-CRE context-specific binding measurements in GRN	- TP = TF-CRE in GRN and in measurement collection - FP = TF-CRE in GRN but not in measurement collection - FN = TF-CRE not in GRN but in measurement collection
	CREs	Presence of annotated CREs in GRN	- TP = CRE in GRN and in annotation - FP = CRE in GRN but not in annotation - FN = CRE not in GRN but in annotation
	Regulated genes	Presence of context-specific CRE with annotated regulation to a neighboring gene in GRN	- TP = CRE-Gene in GRN and in annotation - FP = CRE-Gene in GRN but not in annotation - FN = CRE-Gene not in GRN but in annotation

Figure A.4: **Definition of evaluation metrics.** List of evaluation metrics used to assess the performance of GRNs together with their objective and definition. Metrics that mention context-specific information were tailored for each specific dataset. For example, when working with a heart dataset only entries that contain cardiac-related labels would be used. True Positives (TP), False Positives (FP) and False Negatives (FN). From [5].

A.5 Stability of GRETA methods

All figures in this section are based on runs on the pituitary gland dataset.

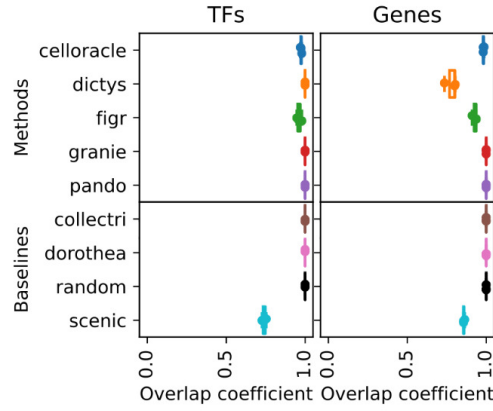


Figure A.5: Overlap coefficients of TFs (left) and target genes (right) between three separate runs per method.

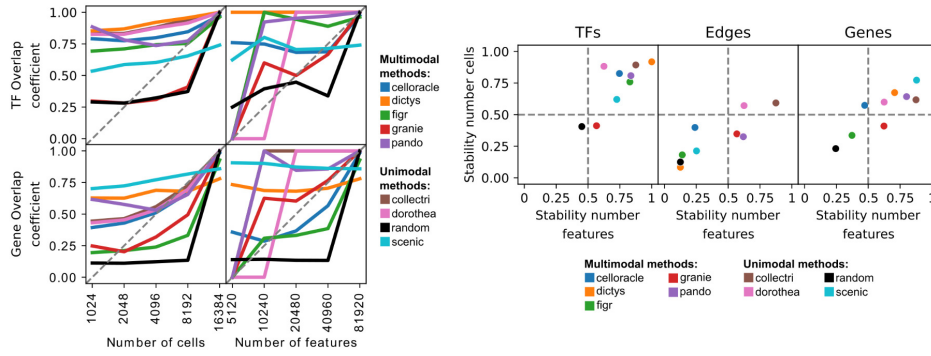


Figure A.6: **On the left:** Overlap coefficients of TFs (top) and target genes (bottom) between GRNs inferred from downsampled datasets with the one inferred using all available data. **On the right:** Stability scores for TFs, edges and target genes. The x -axis indicates the stability score when the number of cells is fixed and features are downsampled, while the y -axis indicates the stability score when the number of features are fixed and cells are downsampled.

A.6 TF Activity Correlation Between LINGER and SCENIC+

Figure A.7 shows the Spearman correlation of TF activity z-scores between LINGER and SCENIC+ across the 54 shared TFs for each cell type. NK cells are excluded as their correlation was not statistically significant ($p = 0.278$).

For the remaining 8 cell types, all correlations are statistically significant ($p < 0.05$), indicating that the two methods capture broadly consistent regulatory signals. The strongest agreement is observed in CD16 Mono ($\rho = 0.84$), while naive T cell subsets (CD4 Naive, $\rho = 0.34$; CD8 Naive, $\rho = 0.29$) show the weakest correlation.

These results indicate that LINGER and SCENIC+ produce concordant TF activity estimates at the global level, with stronger agreement in monocyte populations and weaker agreement in naive T cells.

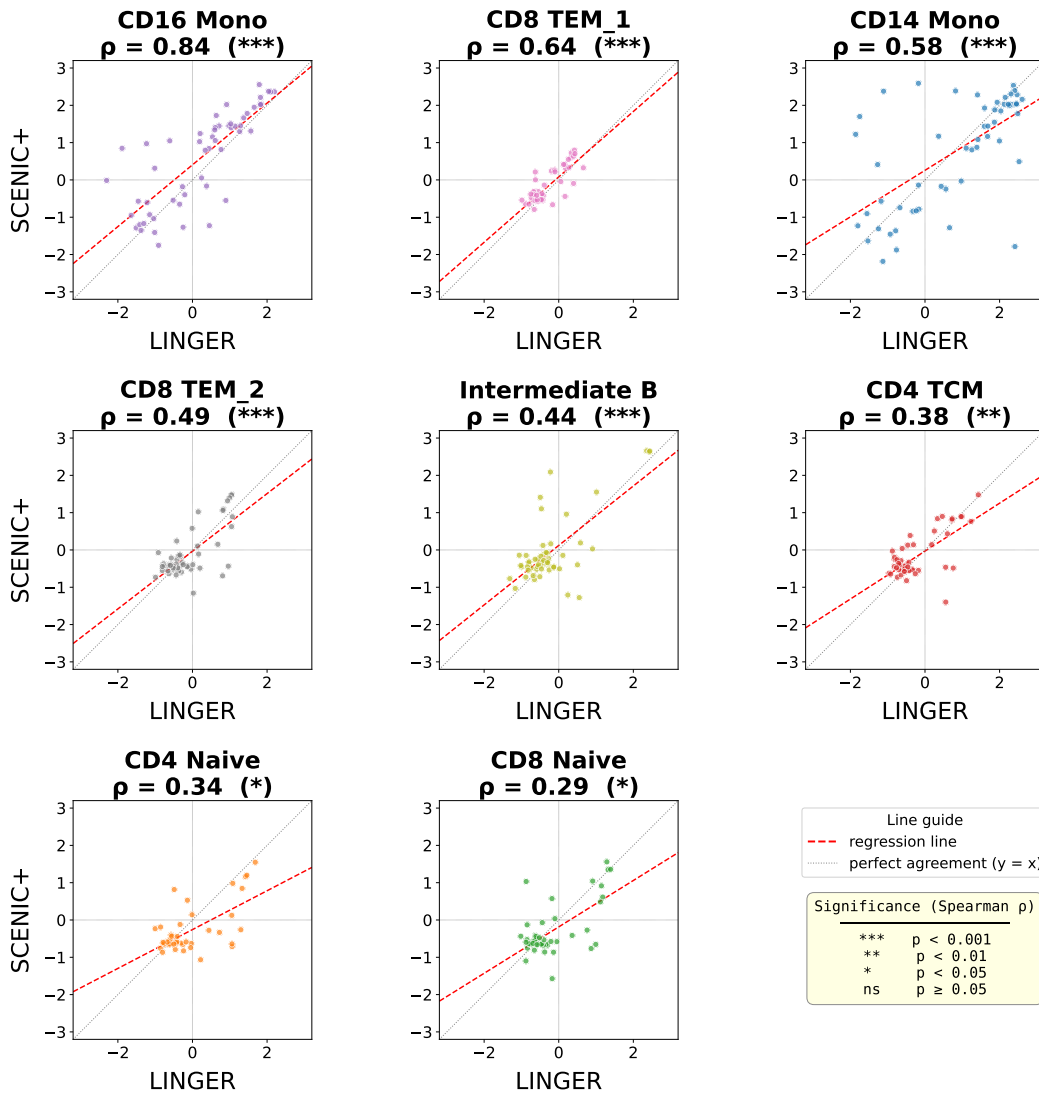


Figure A.7: Spearman correlation of TF activity z-scores between LINGER and SCENIC+ across the 54 shared TFs, shown for each cell type. Each point represents one TF. Cell types are ordered by decreasing correlation *.

Bibliography

- [1] Lawrence Brody. Gene expression. [genome.gov/genetics-glossary/Gene-Expression](https://www.genome.gov/genetics-glossary/Gene-Expression), 2026. Talking Glossary of Genomic and Genetic Terms. Accessed: 2026-05-14.
- [2] Pau Badia-i-Mompel, Lorna Wessels, Sophia Müller-Dott, Rémi Trimbour, Ricardo O. Ramirez Flores, Ricard Argelaguet, and Julio Saez-Rodriguez. Gene regulatory network inference in the era of single-cell multi-omics. *Nature Reviews Genetics*, 24(11):739–754, 2023. [doi:10.1038/s41576-023-00618-5](https://doi.org/10.1038/s41576-023-00618-5).
- [3] Daniel Kim, Andy Tran, Hani Jieun Kim, Yingxin Lin, Jean Yee Hwa Yang, and Pengyi Yang. Gene regulatory network reconstruction: harnessing the power of single-cell multi-omic data. *npj Systems Biology and Applications*, 9(1):51, 2023. [doi:10.1038/s41540-023-00312-6](https://doi.org/10.1038/s41540-023-00312-6).
- [4] Anjun Ma, Xiaoying Wang, Jingxian Li, Cankun Wang, Tong Xiao, Yuntao Liu, Hao Cheng, Juexin Wang, Yang Li, Yuzhou Chang, Jinpu Li, Duolin Wang, Yuexu Jiang, Li Su, Gang Xin, Shaopeng Gu, Zihai Li, Bingqiang Liu, Dong Xu, and Qin Ma. Single-cell biological network inference using a heterogeneous graph transformer. *Nature Communications*, 14(1):964, 2023. [doi:10.1038/s41467-023-36559-0](https://doi.org/10.1038/s41467-023-36559-0).
- [5] Pau Badia-i-Mompel, Roger Casals-Franch, Lorna Wessels, Sophia Müller-Dott, Rémi Trimbour, Yunxiao Yang, Ricardo O. Ramirez Flores, and Julio Saez-Rodriguez. Comparison and evaluation of methods to infer gene regulatory networks from multimodal single-cell data. *bioRxiv*, 2025. [doi:10.1101/2024.12.20.629764](https://doi.org/10.1101/2024.12.20.629764).
- [6] anndata developers. anndata: Annotated data documentation. anndata.readthedocs.io/en/stable/. Official documentation. Accessed: 2025-05-09.
- [7] Adam Frankish et al. GENCODE: reference annotation for the human and mouse genomes in 2023. *Nucleic Acids Research*, 51(D1):D942–D949, 2023. [doi:10.1093/nar/gkac1071](https://doi.org/10.1093/nar/gkac1071).
- [8] Prabarna Ganguly. Transcription. [genome.gov/genetics-glossary/Transcription](https://www.genome.gov/genetics-glossary/Transcription), 2025. Talking Glossary of Genomic and Genetic Terms. Accessed: 2026-05-20.
- [9] National Human Genome Research Institute. Nucleus. [genome.gov/genetics-glossary/Nucleus](https://www.genome.gov/genetics-glossary/Nucleus), 2025. Talking Glossary of Genomic and Genetic Terms. Accessed: 2026-05-20.
- [10] Prabarna Ganguly. Translation. [genome.gov/genetics-glossary/Translation](https://www.genome.gov/genetics-glossary/Translation), 2025. Talking Glossary of Genomic and Genetic Terms. Accessed: 2026-05-20.

- [11] Julie Segre. Promoter. genome.gov/genetics-glossary/Promoter, 2025. Talking Glossary of Genomic and Genetic Terms. Accessed: 2026-05-20.
- [12] John Kimberly. Importance of enhancers and silencers in transcriptional control in eukaryotes. *Bio Med*, 15:626, 2023. doi:10.35248/0974-8369.23.15.626.
- [13] Enrique Balleza, Lucia N. López-Bojorquez, Agustino Martínez-Antonio, Osbaldo Resendis-Antonio, Irma Lozada-Chávez, Yalbi I. Balderas-Martínez, Sergio Encarnación, and Julio Collado-Vides. Regulation by transcription factors in bacteria: beyond description. *FEMS Microbiology Reviews*, 33(1):133–151, 2009. doi:10.1111/j.1574-6976.2008.00145.x.
- [14] Fuchou Tang, Catalin Barbacioru, Yangzhou Wang, Ellen Nordman, Clarence Lee, Nanlan Xu, Xiaohui Wang, John Bodeau, Brian B. Tuch, Asim Siddiqui, Kaiqin Lao, and M. Azim Surani. mrna-seq whole-transcriptome analysis of a single cell. *Nature Methods*, 6(5):377–382, 2009. doi:10.1038/nmeth.1315.
- [15] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. An overview of gene control. In *Molecular Biology of the Cell*. Garland Science, New York, 4th edition, 2002. ncbi.nlm.nih.gov/books/NBK26885.
- [16] Niels Tjoonk. How scatac-seq works. scdiscoveries.com/blog/knowledge/how-scatac-seq-works, 2023. Single Cell Discoveries. Accessed: 2026-05-09.
- [17] MACS3 Project. Call peaks — MACS3 documentation. macs3-project.github.io/MACS/docs/callpeak. Accessed: 2026-05-20.
- [18] Nora M. Al Aboud, Connor Tupper, and Ishwarlal Jialal. Genetics, epigenetic mechanism. ncbi.nlm.nih.gov/books/NBK532999, 2019. StatPearls. Accessed: 2026-05-20.
- [19] Centers for Disease Control and Prevention. Epigenetics. cdc.gov/genomics-and-health/epigenetics, 2024. Accessed: 2026-05-20.
- [20] Douglas Hanahan and Robert A Weinberg. Hallmarks of cancer: the next generation. *Cell*, 144:646–674, 2011. doi:10.1016/j.cell.2011.02.013.
- [21] Thomas J Gonda and Robert G Ramsay. Directly targeting transcriptional dysregulation in cancer. *Nature Reviews Cancer*, 15:686–694, 2015. doi:10.1038/nrc4018.
- [22] Duxin Sun, Wei Gao, Hongxiang Hu, and Simon Zhou. Why 90% of clinical drug development fails and how to improve it? *Acta Pharmaceutica Sinica B*, 12(7):3049–3062, 2022. doi:10.1016/j.apsb.2022.02.002.
- [23] Theodoros Koutsandreas et al. Network-based approaches for drug target identification. *Annual Review of Biomedical Data Science*, 8, 2025. doi:10.1146/annurev-biodatasci-101424-120950.
- [24] Molly Davies, Maeve Boyce, and Eric Conway. Short circuit: Transcription factor addiction as a growing vulnerability in cancer. *Current Opinion in Structural Biology*, 89:102948, 2024. doi:10.1016/j.sbi.2024.102948.

- [25] Kenji Kamimoto et al. Dissecting cell identity via network inference and in silico gene perturbation. *Nature*, 614:742–751, 2023. doi:[10.1038/s41586-022-05688-9](https://doi.org/10.1038/s41586-022-05688-9).
- [26] Niels Tjoonk. How 10x multiome works: A guide. scdiscoveries.com/blog/knowledge/how-10x-multiome-works-a-guide, 2024. Single Cell Discoveries. Accessed: 2026-05-16.
- [27] Luz Garcia-Alonso, Christian H. Holland, Mahmoud M. Ibrahim, Denes Turei, and Julio Saez-Rodriguez. Benchmark and integration of resources for the estimation of human transcription factor activities. *Genome Research*, 29:1363–1375, 2019. doi:[10.1101/gr.240663.118](https://doi.org/10.1101/gr.240663.118).
- [28] Sophia Müller-Dott, Eirini Tsirvouli, Miguel Vazquez, Ricardo O. Ramirez Flores, Pau Badia-i-Mompel, Robin Fallegger, Dénes Türei, Astrid Lægreid, and Julio Saez-Rodriguez. Expanding the coverage of regulons from high-confidence prior knowledge for accurate estimation of transcription factor activities. *Nucleic Acids Research*, 51(20):10934–10949, 2023. doi:[10.1093/nar/gkad841](https://doi.org/10.1093/nar/gkad841).
- [29] Vinay K. Kartha, Fabiana M. Duarte, Yan Hu, Sai Ma, Joyce Y. Kim, Ajay S. Labade, Andrew S. Liu, Soumya Raychaudhuri, Nir Hacohen, Joshua Z. Levin, Jean Fan, and Jason D. Buenrostro. Functional inference of gene regulation using single-cell multi-omics. *Cell Genomics*, 2(9):100166, 2022. doi:[10.1016/j.xgen.2022.100166](https://doi.org/10.1016/j.xgen.2022.100166).
- [30] Carmen Bravo González-Blas, Seppe De Winter, Gert Hulselmans, Nikolai Hecker, Irina Matetovici, Valerie Christiaens, Suresh Poovathingal, Jasper Wouters, Sara Aibar, and Stein Aerts. SCENIC+: single-cell multiomic inference of enhancers and gene regulatory networks. *Nature Methods*, 20(9):1355–1367, 2023. doi:[10.1038/s41592-023-01938-4](https://doi.org/10.1038/s41592-023-01938-4).
- [31] Thomas Moerman, Sara Aibar Santos, Carmen Bravo González-Blas, Jaak Simm, Yves Moreau, Jan Aerts, and Stein Aerts. GRNBoost2 and arboreto: efficient and scalable inference of gene regulatory networks. *Bioinformatics*, 35(12):2159–2161, 2019. doi:[10.1093/bioinformatics/bty916](https://doi.org/10.1093/bioinformatics/bty916).
- [32] L. Wang, N. Trasanidis, T. Wu, et al. Dictys: dynamic gene regulatory network dissects developmental continuum with single-cell multiomics. *Nature Methods*, 20:1368–1378, 2023. doi:[10.1038/s41592-023-01971-3](https://doi.org/10.1038/s41592-023-01971-3).
- [33] Agata Smialowska, Louella Vasquez, Dania Machlab, Markus Ringné, Simon Elsässer, et al. Transcription factor footprinting. nbis-workshop-epigenomics.readthedocs.io, 2025. NBIS Epigenomics Workshop 2025. Accessed: 2026-05-20.
- [34] Robert C McLeay and Timothy L Bailey. Motif enrichment analysis: a unified framework and an evaluation on ChIP data. *Genome Biology*, 11(4):R36, 2010. doi:[10.1186/1471-2105-11-165](https://doi.org/10.1186/1471-2105-11-165).
- [35] Qiuyue Yuan and Zhana Duren. Inferring gene regulatory networks from single-cell multiome data using atlas-scale external data. *Nature Biotechnology*, 43(2):247–257, 2025. doi:[10.1038/s41587-024-02182-7](https://doi.org/10.1038/s41587-024-02182-7).

- [36] ENCODE Project Consortium. An integrated encyclopedia of DNA elements in the human genome. *Nature*, 489:57–74, 2012. doi:[10.1038/nature11247](https://doi.org/10.1038/nature11247).
- [37] Jaime A. Castro-Mondragon et al. JASPAR 2022: the 9th release of the open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 50(D1):D165–D173, 2022. doi:[10.1093/nar/gkab1113](https://doi.org/10.1093/nar/gkab1113).
- [38] E. Wingender, X. Chen, R. Hehl, H. Karas, I. Liebich, V. Matys, T. Meinhardt, M. Prüß, I. Reuter, and F. Schacherer. Transfac: an integrated system for gene expression regulation. *Nucleic Acids Research*, 28(1):316–319, 2000. doi:[10.1093/nar/28.1.316](https://doi.org/10.1093/nar/28.1.316).
- [39] Daniel E Newburger and Martha L Bulyk. UniPROBE: an online database of protein binding microarray data on protein–dna interactions. *Nucleic Acids Research*, 37(suppl_1):D77–D82, 2009. doi:[10.1093/nar/gkn660](https://doi.org/10.1093/nar/gkn660).
- [40] Arttu Jolma, Jian Yan, Thomas Whittington, Jarkko Toivonen, Kazuhiro R Nitta, Pasi Rastas, Ekaterina Morgunova, Martin Enge, Mikko Taipale, Guangyu Wei, et al. DNA-binding specificities of human transcription factors. *Cell*, 152(1-2):327–339, 2013. doi:[10.1016/j.cell.2012.12.009](https://doi.org/10.1016/j.cell.2012.12.009).
- [41] James Kirkpatrick et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114:3521–3526, 2017. doi:[10.1073/pnas.1611835114](https://doi.org/10.1073/pnas.1611835114).
- [42] Benedek Rozemberczki et al. The Shapley value in machine learning. *arXiv preprint arXiv:2202.05594*, 2022. arxiv.org/abs/2202.05594.
- [43] 10x Genomics. PBMC from a healthy donor — granulocytes removed through cell sorting (10k). 10xgenomics.com/datasets/pbmc-from-a-healthy-donor-granulocytes-removed-through-cell-sorting-10-k-1-standard-2-0-0, 2021. Epi Multiome ATAC + Gene Expression dataset. Accessed: 2026-05-11.
- [44] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, et al. RNA velocity of single cells. *Nature*, 560:494–498, 2018. doi:[10.1038/s41586-018-0414-6](https://doi.org/10.1038/s41586-018-0414-6).
- [45] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. Heterogeneous graph transformer. *Proceedings of The Web Conference 2020 (WWW ’20)*, pages 2704–2710, 2020. doi:[10.1145/3366423.3380027](https://doi.org/10.1145/3366423.3380027).
- [46] Yian Chen. Introduction to kullback-leibler divergence. medium.com/@yian.chen261/introduction-to-kullback-leibler-divergence-2d76979d1d8c, 2025. Medium article. Accessed: 2026-05-13.
- [47] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008. doi:[10.1088/1742-5468/2008/10/P10008](https://doi.org/10.1088/1742-5468/2008/10/P10008).
- [48] Johannes Rainer. *EnsDb.Hsapiens.v86: Ensembl based annotation package*, 2017. R package version 2.99.0.

- [49] Snakemake Developers. Snakemake documentation. [snake-make.readthedocs.io/en/stable/](https://snakemake.readthedocs.io/en/stable/), 2026. Official documentation. Accessed: 2026-05-24.
- [50] Sean K. Wang and Howard Y. Chang. Single-cell multiome of human RPE and choroid (GSE262151). ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE262151, 2025. Gene Expression Omnibus dataset. Accessed: 2025-05-09.
- [51] National Center for Biotechnology Information. Gene expression omnibus (GEO). ncbi.nlm.nih.gov/geo/. Official database. Accessed: 2025-05-09.
- [52] Ensembl. Stable identifiers. ensembl.org/info/genome/stable_ids/index.html. Official documentation. Accessed: 2025-05-09.
- [53] Danila Bredikhin, Ilia Kats, and Oliver Stegle. MUON: multimodal omics analysis framework. *Genome Biology*, 23:42, 2022. doi:10.1186/s13059-021-02577-8.
- [54] Isaac Virshup, Sergei Rybakov, Fabian J. Theis, Philipp Angerer, and F. Alexander Wolf. anndata: Access and store annotated data matrices. *Journal of Open Source Software*, 9(101):4371, 2024. doi:10.21105/joss.04371.
- [55] Jonas Simon Fleck, Sophie M. A. Jansen, Damian Wollny, Fabian Zenk, Makiko Seimiya, Aryan Jain, Ryoko Okamoto, Malgorzata Santel, Zhisong He, J. Gray Camp, and Barbara Treutlein. Inferring and perturbing cell fate regulomes in human brain organoids. *Nature*, 2022. doi:10.1038/s41586-022-05279-8.
- [56] Sara Aibar, Carmen Bravo González-Blas, Thomas Moerman, Vân Anh Huynh-Thu, Hana Imrichova, Gert Hulselmans, Florian Rambow, Jean-Christophe Marine, Pierre Geurts, Jan Aerts, Joost van den Oord, Zeynep Kalender Atak, Jasper Wouters, and Stein Aerts. SCENIC: single-cell regulatory network inference and clustering. *Nature Methods*, 14:1083–1086, 2017. doi:10.1038/nmeth.4463.
- [57] OSU-BMBL. DeepMAPS – GitHub repository. github.com/OSU-BMBL/deepmaps. GitHub repository. Accessed: 2025-05-09.
- [58] Satija Lab. Seurat. satijalab.org/seurat/. Official documentation. Accessed: 2025-05-09.
- [59] scVelo developers. scvelo: RNA velocity generalized through dynamical modeling. scvelo.readthedocs.io/en/stable/. Official documentation. Accessed: 2025-05-09.
- [60] Satija Lab. FindMultiModalNeighbors: Weighted nearest neighbor analysis. satijalab.org/seurat/articles/weighted_nearest_neighbor_analysis. Official Seurat documentation. Accessed: 2025-05-09.
- [61] Susanna S. Ng et al. The NK cell granule protein NKG7 regulates cytotoxic granule exocytosis and inflammation. *Nature Immunology*, 21:1205–1218, 2020. doi:10.1038/s41590-020-0758-6.
- [62] NCBI. GZMA granzyme A [Homo sapiens]. ncbi.nlm.nih.gov/gene/3001. NCBI Gene database entry. Accessed: 2025-05-09.

- [63] Samuel A. Lambert, Arttu Jolma, Lisa F. Campitelli, Pratyush K. Das, Yimeng Yin, Mihai Albu, Xiaoting Chen, Jussi Taipale, and Timothy R. Hughes. The human transcription factors. *Cell*, 175(2):598–599, 2018. [doi:10.1016/j.cell.2018.09.045](https://doi.org/10.1016/j.cell.2018.09.045).
- [64] Chenchen Feng, Chao Song, Shuang Song, Guorui Zhang, Mingxue Yin, Yuexin Zhang, Fengcui Qian, Qiuyu Wang, Maozu Guo, and Chunquan Li. KnockTF 2.0: a comprehensive gene expression profile database with knockdown/knockout of transcription (co-)factors in multiple species. *Nucleic Acids Research*, 2024. [doi:10.1093/nar/gkad1016](https://doi.org/10.1093/nar/gkad1016).
- [65] Pau Badia-i-Mompel. Processed KnockTF2 experiments of human cells. doi.org/10.5281/zenodo.7035528, 2022. Zenodo dataset. Accessed: 2025-05-09.
- [66] Pau Badia-i-Mompel et al. decoupler: ensemble of computational methods to infer biological activities from omics data. *Bioinformatics Advances*, 2:vbac016, 2022. [doi:10.1093/bioadv/vbac016](https://doi.org/10.1093/bioadv/vbac016).
- [67] GeeksforGeeks. Benjamini–hochberg procedure. [geeksforgeeks.org/data-science/benjamini-hochberg-procedure/](https://www.geeksforgeeks.org/data-science/benjamini-hochberg-procedure/), 2024. GeeksforGeeks article. Accessed: 2025-05-09.
- [68] CD Genomics. What is gene set enrichment analysis? cd-genomics.com/resource-what-is-gene-set-enrichment-analysis.html. Educational resource. Accessed: 2026-05-25.
- [69] Ronald A. Fisher. On the interpretation of χ^2 from contingency tables, and the calculation of P. *Journal of the Royal Statistical Society*, 85(1):87–94, 1922. [doi:10.2307/2340521](https://doi.org/10.2307/2340521).
- [70] Matthew T. Weirauch et al. Determination and inference of eukaryotic transcription factor sequence specificity. *Cell*, 158(6):1431–1443, 2014. [doi:10.1016/j.cell.2014.08.009](https://doi.org/10.1016/j.cell.2014.08.009).
- [71] Zidong Zhang et al. Single nucleus transcriptome and chromatin accessibility of postmortem human pituitaries reveal diverse stem cell regulatory mechanisms. *Cell Reports*, 38:110467, 2022. [doi:10.1016/j.celrep.2022.110467](https://doi.org/10.1016/j.celrep.2022.110467).
- [72] Arthur Liberzon, Ceren Birger, Helga Thorvaldsdóttir, Mahmoud Ghandi, Jill P. Mesirov, and Pablo Tamayo. The molecular signatures database (MSigDB) hallmark gene set collection. *Cell Systems*, 1(6):417–425, 2015. [doi:10.1016/j.cels.2015.12.004](https://doi.org/10.1016/j.cels.2015.12.004).
- [73] Michael Schubert, Bernd Klinger, Marcel Klünemann, Alexander Sieber, Florian Uhlitz, Sebastian Sauer, Mathew J. Garnett, Nils Blüthgen, and Julio Saez-Rodriguez. Perturbation-response genes reveal signaling footprints in cancer gene expression. *Nature Communications*, 9(1):20, 2018. [doi:10.1038/s41467-017-02391-6](https://doi.org/10.1038/s41467-017-02391-6).
- [74] The Human Protein Atlas. The human immune cells – t-cells. proteinatlas.org/humanproteome/single+cell/immune+cell/t-cells. Human Protein Atlas immune cell resource. Accessed: 2025-05-09.

- [75] Christopher Pratt. A guide to naïve t-cell markers. biocompare.com/Editorial-Articles/597618-A-Guide-to-Naive-T-Cell-Markers/, 2023. Biocompare article. Accessed: 2025-05-09.
- [76] Arnaud Delpoux, Chen-Yen Lai, Stephen M. Hedrick, and Andrew L. Doedens. Foxo1 opposition of cd8+ t cell effector programming confers early memory properties and phenotypic diversity. *Proceedings of the National Academy of Sciences*, 114(42):E8865–E8874, 2017. doi:10.1073/pnas.1618916114.
- [77] A. F. Gülich, R. Rica, C. Tizian, C. Viczenczova, K. Khamina, T. Faux, D. Hainberger, T. Penz, R. Bosselut, C. Bock, A. Laiho, L. L. Elo, A. Bergthaler, W. Ellmeier, and S. Sakaguchi. Complex interplay between mazz and runx3 regulates the generation of cytotoxic t lymphocyte and memory t cells. *Frontiers in Immunology*, 12:535039, 2021. doi:10.3389/fimmu.2021.535039.
- [78] R. Revilla-I-Domingo, I. Bilic, B. Vilagos, H. Tagoh, A. Ebert, I. M. Tamir, L. Smeenk, J. Trupke, A. Sommer, M. Jaritz, and M. Busslinger. The b-cell identity factor pax5 regulates distinct transcriptional programmes in early and late b lymphopoiesis. *EMBO Journal*, 31(14):3130–3146, 2012. doi:10.1038/emboj.2012.155.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN

École polytechnique de Louvain

Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | www.uclouvain.be/epl