UNIVERSITÀ DI TRENTO
DIPARTIMENTO DI MATEMATICA
PH.D SCHOOL IN MATHEMATICS

INFERENCE OF GENE REGULATORY NETWORKS WITH INTEGRATION OF PRIOR KNOWLEDGE

Author:
Emiliano Maresi

Supervisor:
Prof. Mario Lauria

XXXV Cycle
A.A/A.Y. 2023/2024

Abstract

Gene regulatory networks (GRNs) are crucial for understanding complex biological processes and disease mechanisms, particularly in cancer. However, GRN inference remains challenging due to the intricate nature of gene interactions and limitations of existing methods. Traditionally, prior knowledge in GRN inference simplifies the problem by reducing the search space, but its full potential is unrealized. This research aims to develop a method that uses prior knowledge to guide the GRN inference process, enhancing accuracy and biological plausibility of the resulting networks.

We extended the Fused Sparse Structural Equation Models (FSSEM) framework to create the Fused Lasso Adaptive Prior (FLAP) method. FSSEM incorporates gene expression data and genetic variants in the form of expression quantitative trait loci (eQTLs) perturbations. FLAP enhances FSSEM by integrating prior knowledge of gene-gene interactions into the initial network estimate, guiding the selection of relevant gene interactions in the final inferred network.

We evaluated FLAP using synthetic data to assess the impact of incorrect prior knowledge and real lung cancer data, using prior knowledge from various gene network databases (GIANT, TissueNexus, STRING, ENCODE, hTFtarget). Our findings demonstrate that integrating prior knowledge improves the accuracy of inferred networks, with FLAP showing tolerance for incorrect prior knowledge. Using real lung cancer data, functional enrichment analysis and literature validation confirmed the biological plausibility of the networks inferred by FLAP. Different sources of prior knowledge impacted the results, with GIANT providing the most biologically relevant networks, while other sources showed less consistent performance.

FLAP improves GRN inference by effectively integrating prior knowledge, demonstrating robustness against incorrect prior knowledge. The method’s application to lung cancer data indicates that high-quality prior knowledge sources like GIANT enhance the biological relevance of inferred networks. Future research should focus on improving the quality and integration of prior knowledge, possibly by developing consensus methods that combine multiple sources. This approach has potential applications in cancer research and drug sensitivity studies, offering a more accurate understanding of gene regulatory mechanisms and potential therapeutic targets.

Table of Contents

Chapter 1: INTRODUCTION

Gene regulatory networks (GRNs) are complex systems of interconnected genes within cells, governing the regulation of gene expression and coordinating various cellular processes. These networks represent the causal relationships among genes, describing how the activation or inhibition of one gene can influence the expression of others.

Understanding the structure of GRNs is particularly valuable in life sciences and systems biology. It helps reveal the biological mechanisms driving cellular functions and enables the study of complex diseases like cancer. In these diseases, dysfunction is not solely dependent on individual genes but on the rewiring of gene interactions, which drive pathological processes.

Understanding the biological mechanisms and dysfunctions within cells begins with comprehending the cell state. High-throughput technologies enable the capture of snapshots of the cell state, providing data like gene expression, protein abundance, genetic variations, etc. Through bioinformatic methods, it is possible to analyze and interpret these omics data to extract meaningful evidence. This information can then be used to infer GRNs by reverse-engineering through the construction of mathematical models which allow to retrieve the structure of gene interactions that generated the observed data.

Although the problem of GRN inference has gained attention for the past twenty years, the absence of a comprehensive method proficient in all aspects of the task underscores the complexity of GRN inference. As a result, existing methods frequently target specific aspects or subsets of the problem.

Over the years, the Dialogue for Reverse Engineering Assessment and Methods (DREAM) challenges [1], [2], [3], [4] have served as a benchmarking competition for gene regulatory network inference methods. These challenges unveiled the limitations of methods relying solely on gene expression data, prompting subsequent research efforts to develop state-of-the-art inference methods that integrate multiple omic data, perturbations, and prior knowledge. Among the best performing methods, Structural Equation Models (SEMs) became a successful framework to infer GRNs using multi omics and perturbations. In particular Fused Sparse SEM (FSSEM) [5] is able to infer two GRNs for paired datasets that share similar gene regulations, i.e. tumor samples vs healthy samples.

While the integration of multiple omics data and perturbations has a clear definition and application, the same cannot be said for prior knowledge. Prior knowledge, referring to a priori information assumed to hold some degree of truth, is incorporated variably depending on the specific assumptions and implementation of each method.

In our review of state-of-the-art methods, we observed that prior knowledge is primarily utilized to reduce the complexity of the problem. Each method employs its own set of prior knowledge, reflecting different assumptions and strategies for addressing the complexity of gene regulatory networks.

In contrast to existing approaches, we questioned whether prior knowledge could be used to guide and enhance the inference process rather than merely reducing its complexity. In our project, we explored a novel application of prior knowledge, aiming to enrich the inference process. Specifically, we investigated how prior knowledge could be effectively integrated with omics data to guide the inference process, acknowledging the imperfection inherent in prior knowledge.

From this exploration, we developed Fused Lasso Adaptive Prior (FLAP), a GRN inference method designed to leverage prior knowledge of the GRN structure to guide the inference process and improve its accuracy. FLAP integrates prior knowledge to enhance the inference process and steer it toward more biologically plausible results.

This thesis is structured in the following way: Chapter 1 introduces GRN inference, providing a review of the methods developed over the years, from which we derive the motivations for our work. Chapter 2 describes in detail the use of Structural Equation Modeling (SEM) methods applied to GRN inference, along with the linear models used to solve them. We also explain FSSEM and how our FLAP method extends it by incorporating prior knowledge. In Chapter 3, we test FLAP on synthetic data to calibrate the integration of prior knowledge. Subsequently, we test FLAP on real data along with prior knowledge retrieved from databases, validating the plausibility of the inferred GRNs. Chapter 4 provides a summary of our work key points and future directions.

1.1 Biological background

The biology of living organisms can be described by the central dogma of molecular biology, which outlines the flow of genetic information within a biological system: information is stored in DNA in the form of genes. Genes are expressed through their transcription into mRNA, which is subsequently translated into proteins. These proteins perform various biological functions that are essential for maintaining life by controlling biochemical reactions, regulating the levels of compounds, and more.

These biological processes are fundamentally based on gene interactions. In particular, transcription factors (TFs) are proteins that help turn specific genes 'on' or 'off' by recognizing and binding to specific DNA sequences, known as DNA motifs, located in the promoter regions of target genes. By binding to these motifs, TFs can either activate or repress the transcription of target genes, thereby modulating their expression levels.

The interactions between genes and transcription factors form complex networks of regulatory relationships. These networks, known as gene regulatory networks (GRNs), define the structure and dynamics of gene expression within a cell.

Reconstructing GRNs is especially relevant in the life sciences, such as medicine and biological research, because it can reveal the underlying rules of conditions affecting cells, such as metabolic dysfunction, cancer replication, and drug sensitivity or resistance.

However, the direct observation of gene interactions within organisms is often impractical. To address this challenge, researchers turn to high-throughput technologies that allow for the retrieval of large-scale measurements of the components of these regulatory processes, such as DNA, mRNA, proteins, and metabolites.

When these measurements are considered individually, they are referred to as single omics data. Genomics pertains to DNA, transcriptomics refers to the gene expression levels of mRNA, proteomics pertains to the abundance of proteins, and metabolomics refers to the concentration of metabolites. When single omics measurements are integrated, they form multi-omics datasets, which provide a more comprehensive view of biological systems by considering different molecular layers.

The task of reverse-engineering a GRN from high-throughput data is a hot topic in the fields of bioinformatics and systems biology. Reverse-engineering a GRN involves inferring the underlying network structure that governs gene interactions from experimental data. This process uses computational and mathematical approaches to model complex gene interactions, aiming to describe the regulatory mechanisms that generate the observed gene expression patterns.

1.1.1 Types of Data

To construct gene regulatory networks, various types of biological data can be utilized, each providing unique insights into the regulatory processes within cells. This section will focus on the primary types of data used in this thesis, while also acknowledging other relevant data types in the field.

Primary data types used in this thesis:

Gene Expression Data:

Microarray Data: This technology measures the expression levels of thousands of genes simultaneously by hybridizing cDNA to DNA probes fixed on a solid surface. It provides a snapshot of gene expression under specific conditions and is valuable for identifying co-expression patterns and differential gene expression.

RNA Sequencing (RNA-seq): RNA-seq offers a detailed and quantitative measurement of gene expression by sequencing cDNA derived from RNA samples. It captures the entire transcriptome, including rare and novel transcripts, providing high-resolution data on gene expression levels.

Single-cell RNA Sequencing (scRNA-seq): scRNA-seq measures gene expression at the single-cell level, allowing for the analysis of cellular heterogeneity and the identification of distinct cell populations within a sample.

Genomic Data:

Single Nucleotide Polymorphisms (SNPs): SNP microarrays reveal genetic variations at specific nucleotide positions in the genome. These variations can influence gene regulation and expression. Specifically, it is possible to focus on condition-specific or tissue-specific GRNs by using a combination of transcriptomics data (gene expression levels) and genomics data, which contains information about SNPs, variations of single DNA bases at specific loci in the genome. These genetic variations can affect gene expression by altering non-coding regulatory elements of genes, such as promoters and enhancers, and are referred to as expression quantitative trait loci (eQTLs). eQTLs can be categorized as cis-eQTLs when they are close to the gene they regulate,

typically within 1 megabase (1 Mb), or trans-eQTLs when they are located far from the gene or on a different chromosome. Usually, cis-eQTLs are more informative since their effect on gene expression is stronger, and thus trans-eQTLs are often not considered.

Epigenomic Data:

Chromatin Immunoprecipitation Sequencing (ChIP-seq): ChIP-seq identifies binding sites of DNA-associated proteins, such as transcription factors and histone modifications, across the genome. This data helps map regulatory elements and understand the mechanisms of gene regulation.

ChIP on Chip: This technique combines chromatin immunoprecipitation with microarray technology to identify protein-DNA interactions and histone modifications. It provides insights into the regulatory regions of the genome.

Gene Perturbation Data:

RNA Interference (RNAi): RNAi is used to knock down gene expression by degrading mRNA transcripts. Perturbation of gene expression through RNAi can help identify the functional roles of specific genes and their regulatory interactions.

CRISPR/Cas9: This technology is used for gene knockout (KO), allowing precise deletion of gene function. CRISPR/Cas9-induced gene perturbations help elucidate gene functions and regulatory networks.

CRISPR Interference (CRISPRi): CRISPRi uses a catalytically inactive Cas9 (dCas9) fused to a repressor domain to specifically inhibit gene transcription, resulting in gene knockdown. It is a powerful tool for studying gene regulation by selectively repressing target genes.

Gene Overexpression: This involves increasing the expression of a gene to study its effect on cellular processes. Overexpression studies help determine the impact of elevated gene activity and identify regulatory interactions.

Other Data Types Used in the Field:

While not the primary focus of this thesis, other types of omics data can also be instrumental in constructing GRNs:

Proteomic Data: Quantitative measurements of protein levels provide insights into the functional state of the cell. Proteomics can reveal post-transcriptional regulatory mechanisms and protein-protein interactions.

Metabolomic Data: Metabolomics involves the analysis of metabolites within a biological sample. Metabolite profiles reflect the biochemical activity and metabolic state of cells, providing additional information for GRN construction.

Chromatin Accessibility Data: Techniques like ATAC-seq identify regions of open chromatin, indicating active regulatory regions. This data can be linked to gene expression to infer regulatory elements and interactions.

1.2 Mathematical Definition of a Network

To understand the structure and analysis of Gene Regulatory Networks (GRNs), it is essential to define the mathematical framework used to represent these networks. A network, in the context of biological systems, is a collection of nodes (representing genes, proteins, or other molecules) and edges (representing interactions between these nodes).

A network can be mathematically represented as a graph $G = (V, E)$, where V is a set of nodes (vertices), $V = \{v_1, v_2, \ldots, v_n\}$, and E is a set of edges, $E \subseteq V \times V$, where each edge represents an interaction between nodes.

Graphs can be categorized based on the type of edges they contain. In an undirected graph, an edge is defined as a two-element subset of the set of nodes V. This means the edge does not have a direction. We can describe an undirected edge as an undordered pair of vertices $e = \{v_i, v_j\}$, which connects node v_i with node v_j. Here, $\{v_i, v_j\} = \{v_j, v_i\}$, indicating

the bidirectional nature of the interaction. Undirected edges are typically drawn as simple lines connecting the nodes.

In a directed graph, an edge is defined as an ordered pair of nodes. This means the edge has a direction, indicating the interaction goes from one node to another. We describe a directed edge as an ordered pair $e = (v_i, v_j)$, where node v_i precedes node v_j. Here, $(v_i, v_j) \neq (v_j, v_i)$, reflecting the unidirectional nature of the interaction.
Directed edges are typically drawn as arrows pointing from the source node v_i to the target node v_j.

An adjacency matrix A is a square matrix used to represent a graph. The elements of the matrix indicate whether pairs of nodes are adjacent (i.e., directly connected) in the graph. For a graph with n nodes, the adjacency matrix A is an $n \times n$ matrix defined as follows:

$$A_{ij} = \begin{cases} 1 & \text{if there is an edge from node } v_i \text{ to node } v_j \\ 0 & \text{otherwise} \end{cases}$$

For undirected graphs, the adjacency matrix is symmetric ($A_{ij} = A_{ji}$).
For directed graphs, the adjacency matrix is not necessarily symmetric ($A_{ij} \neq A_{ji}$)

In some networks, edges are assigned weights to represent the strength, capacity, or other attributes of the interactions between nodes. A weighted edge between nodes v_i and v_j can be represented as (v_i, v_j, w_{ij}), where w_{ij} is the weight of the edge. In the adjacency matrix, this is reflected as:

$$W_{ij} = \begin{cases} w_{ij} & \text{if there is an edge from node } v_i \text{ to node } v_j \text{ with weight } w_{ij} \\ 0 & \text{otherwise} \end{cases}$$

Weighted edges provide more information than unweighted edges and are essential in many applications, such as modeling the strength of gene interactions in gene regulatory networks.

A path in a graph is a sequence of edges that connects a sequence of distinct nodes. Formally, a path P from node v_i to node v_j is represented as:

$$P=\{(v_i,v_k),\ (v_k,v_l),\ \ldots,\ (v_m,v_j)\}$$

In the context of gene regulatory networks, a pathway represents a series of regulatory interactions through which genes influence one another. For example, a pathway might involve a sequence of transcription factors that activate or repress a series of target genes, ultimately controlling a biological process.

A cycle in a graph is a path that starts and ends at the same node without traversing any edge more than once. Formally, a cycle C is defined as a path where the starting node and the ending node are the same, i.e.,

$$C=\{(\boldsymbol{v_i},v_k),\ (v_k,v_l),\ \ldots,\ (v_m,\boldsymbol{v_i})\}$$

A graph that contains at least one cycle is called a cyclic graph. These graphs are important in biological systems where feedback loops are common, such as in metabolic networks or regulatory circuits. A graph that contains no cycles is called an acyclic graph. When a directed graph is acyclic, it is referred to as a Directed Acyclic Graph (DAG). DAGs are particularly important in modeling hierarchical relationships, such as those found in certain gene regulatory networks where genes are regulated in a cascading manner without feedback loops.

The degree of a node in a network describes the number of connections a node has with other nodes. The degree of a node v_i , denoted as $deg(v_i)$, is the number of edges connected to v_i

$$deg(v_i) = \sum_j A_{ij}$$

where A is the adjacency matrix of the graph, and A_{ij} indicates the presence (1) or absence (0) of an edge between nodes v_i and v_j .

1.3 Review of gene regulatory network inference methods

1.3.1 Early methods for gene regulatory network inference using gene expression data

In the early days of systems biology, researchers sought to unravel the complexities of gene regulatory networks using gene expression data as their primary source of information. With the advent of high-throughput technologies such as microarrays and later, RNA sequencing, large-scale gene expression data became readily available, sparking the development of computational methods for gene network inference.

In this section, we explore some of the early methods used for gene network inference, their underlying principles, and their limitations.

Correlation and relevance networks:

Early methods for inferring gene regulatory networks from gene expression data constructed their networks using co-expression similarity measures based on correlation coefficients (such as Pearson's or Spearman's) or Mutual Information-based measures. These approaches are broadly categorized into correlation networks and relevance networks.

Correlation networks utilize co-expression similarity measures based on correlation coefficients to identify relationships between genes based on their expression patterns. Methods like Weighted correlation network analysis (WGCNA), partial correlation, and Gaussian Graphical Models fall into this category.

Relevance networks, however, rely on Mutual Information-based measures to identify direct regulatory relationships between genes, without assuming linear relationships. Approaches like Algorithm for the Reconstruction of Accurate Cellular Networks (ARACNE) [6] , Context Likelihood of Relatedness (CLR) [7] and Minimum Redundancy NETwork (MRNET) [8] , conservative causal core (C3NET) [9] belong to this category.

Correlation and relevance networks are symmetrical, with undirected edges. Causal relationships between genes can only be assumed if the regulator genes are known in advance. Their primary use is to explore the co-regulation of genes. These networks are often employed in combination with clustering approaches to identify coherent gene modules.

Bayesian Networks:

Bayesian Networks were among the first methods to allow the inference of gene regulatory networks where edges represented putative causal dependencies between genes [10][11].
In this network, the structure is represented by a directed acyclic graph (DAG) where genes are random variables drawn from conditional probability distribution where there is a set of parents for each node.

This structure defines the decomposition of the joint distribution over all random variables into the conditional distribution of each gene. It is based on the Markov assumption, which states that each gene is independent of its non-descendants.

The inference process consists of two parts. Firstly, there's model selection, which aims to identify the network structure that best explains the observed data. This is done using Bayesian scoring metrics such as Akaike's Information Criterion (AIC) and Bayesian Information Criterion (BIC) to penalize complex models and select the simplest ones [12]. Secondly, there's parameter learning. This part estimates the probabilities associated with each gene in the network.

The fundamental limitation of Bayesian Networks lies in model selection. The number of topologies increases super-exponentially with the number of genes, making it infeasible to compute the likelihood of all networks. For this reason, they are applicable only to small networks. However, this limitation can be partly compensated for by using heuristics or locally constrained search techniques. Another limitation is the inability to model cycles (i.e., feedback loops). This issue is addressed in dynamic Bayesian networks [13].

Regression based models:

Regression methods are widely employed in gene regulatory network (GRN) inference due to their ability to model the relationships between gene expression levels. In gene regulatory networks, genes often regulate the expression of other genes, and regression methods offer a means to quantify and comprehend these regulatory relationships. By considering the expression level of a gene as the dependent variable and the expression levels of its regulator genes as independent variables, regression methods allow us to describe how changes in the expression levels of regulator genes influence the expression level of the target gene.

Linear regression models are commonly utilized for this analysis due to their simplicity. These models assume that the relationship between the expression levels of genes can be captured by a

linear equation. In this equation, the expression level of the target gene is a linear combination of the expression levels of its regulator genes, with some noise.

Fitting a regression model to gene expression data enables the identification of regulator genes that are most strongly associated with the expression of a target gene. Since each gene is typically regulated by only a small number of other genes [14] [15], regularized linear regression methods reframe the inference problem as a feature selection problem. Feature selection strategies are stepwise selection [16] , least angle regression [17] , ridge regression [18] and least absolute shrinkage and selection operator (LASSO) [19] . Among these methods, LASSO has become the most popular in gene network inference, due to its ability to shrink less relevant coefficients to set them to exactly zero, thus leading to a sparser linear model with fewer predictors per target gene.

Another category of regression approaches for GRN inference includes tree-based ensemble regression methods. In contrast to regularized linear regression, these methods do not make assumptions about gene regulation, allowing them to infer both linear and non-linear interactions.

Random forest regression, exemplified by methods like GENIE3 [20] , is a tree-based approach. In this method, the gene expression profile dataset is bootstrapped over samples, and a decision tree is constructed over each bootstrapped dataset. This process results in separate rankings of genes as potential regulators of a target gene. The final network is obtained by averaging the rankings over all the decision trees. While this approach shares the concept of selecting a set of genes as potential regulators with linear regression, it differs in that the user defines the maximum rank threshold to include regulators that best explain the target gene expression profile.

The main limitation of linear models is their effectiveness in situations where the gene expression experiment is conducted within a slowly changing system or around a steady state. Linear models tend to provide more accurate predictions when applied to such data, as they may struggle to capture the dynamics of gene regulatory networks in rapidly changing or non-steady state conditions [21] .

Differential equation models:

Approaches based on differential equations represent regression models aim to emulate the biological mechanism of transcriptional regulation [22][23][24] . They are based on systems of ordinary differential equations (ODEs) for deterministic modeling and stochastic differential equations (SDEs) for stochastic modeling. A gene network is described by a set of first-order differential equations, which detail the rate of change of the gene expression of a target gene as a function of the expression profiles of other genes. In general, differential equations are a

combination of nonlinear functions because all concentrations become saturated at some point in time.

ODE methods can be highly computationally demanding since they model multiple solutions to explain observed expression profiles fluctuations in the data. Introducing constraints, such as known kinetic parameters or prior knowledge of GRN structure, can significantly benefit ODE-based methods.

1.3.2 DREAM challenges

The Dialogue on Reverse Engineering Assessment and Methods (DREAM) challenges [1][2][3], [4] represent an important contribution to the research of GRN inference methods. DREAM was a series of community-based open challenges aimed at comparing the existing GRN inference methods with a standardized evaluation and assess the best performing methods, understanding their advantages, limitations, and biases. The knowledge acquired in the challenge enabled researchers to choose the best tool to use to address a specific problem.

This need for benchmarking came from the lack of experimental validation of the inferred networks, which were often tested on synthetic datasets or on real datasets in specific scenarios.
The series of DREAM challenges used in silico datasets, and real datasets of E.coli (prokaryotic), S.cervisae (eukaryotic) and S.aureus (human pathogen). For the benchmarks in silico networks were compared to their gold standard network that generated the gene expression dataset, while for the real data they either used well known networks, like for E.coli, or the RegulonDB database.

The DREAM5 [4] was especially important for methods that inferred causal gene regulatory networks because it allowed to benchmark 35 different methods, grouped in six categories: Regression, Mutual Information, Correlation, Bayesian Networks, Other approaches and Meta predictors. Overall, no category outperformed all the other ones, because withing each method there was a mix of poorly and well performing methods, and even among the well performing the precision levels achieved were incredibily low. Of particular interest it was observed that among all the categories, methods that made explicit use of direct transcription factor perturbations (knockout or overexpression) or used the information about TF-binding sites improved their prediction accuracy for downstream targets. This founding led to the more recent GRN inference approaches which integrated perturbations or chromatin immunoprecipitation data with gene expression.

1.3.3 Multi omic and perturbation methods

Learning from the conclusion of the DREAM5 challenge the new GRN inference methods developed in two directions, the first was to use multi omics by using high throughput data of gene expression and chromatin immunoprecipitation (ChIP-chip or ChIP-seq) and the second was to use perturbations and the knowledge of the perturbation design (which genes were perturbed).

Multi omics methods:

Multi omics methods, use chromatin immunoprecipitation (ChIP) to identify binding sites of DNA-associated proteins near genes that are regulated by transcription factors.
This information is combined with databases of TFs binding motifs (i.e. Joint Analysis of Sequence Profiles for Unbiased Recognition of Transcription Factor Binding Sites (JASPAR) [25] , TRANScription FACtor database (TRANSFAC) [26], Encyclopedia of DNA Elements (ENCODE) [27] , ChIP-X Enrichment Analysis (CHEA) [28] to construct a connectivity network of all the possible TF-target regulations.

Then this connectivity network is used as a prior knowledge for the inference methods.
Inferelator3.0 [29] performs a linear regression of the gene expression matrix against the connectivity network to estimate a matrix of coefficients called transcription factor activity (TFA) matrix, which should reflect the latent activity of the transcription factors, then uses clustering algorithms to group TF-target edges, keep the edges in the cluster with the best score and discard the rest to obtain the GRN. CellOracle [30] instead uses Bayesian bagging to select relevant connections. SCENIC [31] uses the connectivity network as a set of candidate TF-target edges, which are scored based on their area under the recovery curve (AUC) enrichment of all genes, then follows a clustering step on each sample/cell and select for each cluster a GRN consisting of the reoccurring edges. MERLIN [32] which extends the expression based GRNs inference algorithm MERLIN, a Bayesian framework of learning a probabilistic graphical model integrating additional structure prior such as sequence-motifs, ChIP data or gene knockout expriments PANDA [33] integrates the prior connectivity networks together with another prior protein-protein interaction (PPI) network from STRING [34] using gene expression. This is used to identify co-regulated transcription factors and co-expressed target genes.

Those methods have two main limitations, the first is that the accessibility to a DNA motif does not necessarily imply the binding of a transcription factors since regulation often involve complexes

of multiple transcription factors that may not be affecting the system, and second, the identification of TF-target regulations relies on database annotations, which may be incorrect.

Perturbation methods:

The DREAM5 challenge showed that the use of known-target perturbations, also called perturbation design, allows inference methods to achieve higher accuracy.

In the work by Seçilmiş 2022 [35] such GRN inference methods where further evaluated. Benchmarks showed that even the worst method that uses perturbation design outperform those that do not, like GENIE3 and TIGRESS which ranked first and second place in the DREAM5 challenge.

However, a few limitations were identified. These methods cannot function without a perturbation design, and their performance can deteriorate easily if the perturbation design is incorrect. This can happen either because the perturbation failed to work, or due to the high noise level, making the perturbation signal difficult to detect in genes downstream of the perturbed gene.

1.3.4 Methods using gene expression and genetic variants

To address the limitations of perturbation experiments, such as knockdown and knockout studies, new gene regulatory network inference methods leverage naturally occurring genetic variations. These methods operate under the assumption that genetic makeup influences transcription levels. For instance, single nucleotide polymorphisms (SNPs) can affect the regulatory regions of genes, giving rise to what is known as expression quantitative trait loci (eQTLs). The primary advantage of using eQTLs is that genetic variants are free from external confounders, making them ideal for inferring causal relationships within gene regulatory networks.

Among those methods, Bayesian networks incorporating eQTLs [36] , likelihood test approaches QDG [37] , and methods that rely on Structural Equation Model (SEM) were developed. The first to use SEM for gene network reconstruction using only gene expression data were Xiong et al. (2004) [38], followed by work such as Encompassing Directed Network (EDN) [39] , which incorporated gene expression and eQTLs.

More recently, Logsdon et al. (2010) [40] improved EDN by incorporating Adaptive Lasso linear regression, an enhanced version of regularized LASSO regression. This work was followed by

further improvements with the introduction of Sparse Maximum Likelihood [41] (Cai et al, 2013), Fused Sparse SEM (FSSEM) [5] and BFDSEM [42] .

FSSEM is particularly relevant as one of the state-of-the-art methods capable of jointly estimating two networks (i.e., case vs. control, tumor vs. normal) and optimizing the inference to identify the differential network. This optimization highlights differential gene regulations, making FSSEM a powerful tool for deciphering gene regulatory networks in the context of different conditions or disease states.

1.3.5 Repositories of gene networks

In addition to inference methods, there are databases that provide resources for the study of GRNs. The first type are the transcription factors (TFs) databases, which collect TFs binding motifs like ENCODE, TRANSFAC, CHEA and JASPAR. These TF-motifs are used by multi omics GRN inference methods to pair the expression profiles of transcription factors with those of their potential target genes that present the DNA motif in their sourrounding region.

Search Tool for the Retrieval of Interacting Genes/Proteins (STRING) is a database of protein-protein interaction networks for a wide range of organisms, from bacteria to humans. It integrates several types of data sources, including experimental data, curated databases (Biological General Repository for Interaction Datasets (BioGRID) [43] , Kyoto Encyclopedia of Genes and Genomes (KEGG) [44] , Reactome Pathway Database (Reactome) [45] , Gene Ontology (GO) [46] , Molecular INTeraction Database (MINT) [47] , and Human Protein Reference Database (HPRD) [48]), text mining of scientific literature (Online Mendelian Inheritance in Man (OMIM) [49] , PubMed), and computational predictions.

Harmonizome 3.0 [50] is a web-based tool designed to explore the functional relationships between genes and proteins. It serves as a collector of genomic databases (i.e. TRANSFAC, KEGG, MotifMap [51] , Molecular Signatures Database(MsigDB) [52]), aggregating and standardizing functional genomics data. Users can explore gene functions, protein-protein interactions, and gene-disease associations through this platform.

The Genome-wide Integrated Analysis of gene Networks in Tissues (GIANT) [53] database is a collection of 144 tissue-specific functional gene networks. These are constructed using a data-driven Bayesian integration method that incorporates a collection of datasets from 14,000 publications while automatically assessing their relevance to each of the 144 tissues and cell lineage-specific functional contexts.

TissueNexus [54] is a database of tissue-specific functional gene networks created using the XGBoost machine learning method to integrate RNA-seq gene expression data from Genotype-Tissue Expression (GTEx) [55] , cancer gene expression patterns from The Cancer Genome Atlas (TCGA), regulatory elements like transcription factors and enhancer regions from ENCODE and NIH Roadmap Epigenomics, functional annotations from Gene Ontology and pathway information from the KEGG database. This database comprises of 49 tissue-specific human gene networks.

1.4 Motivations and objectives

In our review of methods, we observed that despite the numerous research efforts invested over the last twenty years, inferring gene regulatory networks remains a significant challenge. As demonstrated in the DREAM5 challenge, the performance of these methods heavily depends on their implementation rather than the specific approach category.

In addition, the DREAM5 challenge revealed that no single gene regulatory network inference method outperformed the others. This highlights the limitations inherent in inference methods, as they tend to confine the problem to the parts they can handle well with their respective methods.

Building on lessons learned from DREAM5, recent methods have incorporated multi-omics data, gene expression perturbations, and prior knowledge to enhance the inference process. Moreover, the latest models have focused on expanding the size of inferred networks, with SEM employing regularized linear regression emerging as the state-of-the-art data-driven method.

We observed that while the use of multi-omics data and perturbations has a clear definition and application, the utilization of prior knowledge depends on the assumptions and implementation of each method. Generally, prior knowledge refers to "a priori" information with some degree of truth, mainly used to reduce the complexity of the problem. In the methods we reviewed, prior knowledge was employed to define a set of candidate regulators, identify genes that received perturbations, or establish a fixed “seed” network of transcription factors with their target genes from which the inference method selected the best-scoring edges.

Then, we questioned whether it could be possible to use prior knowledge of gene interactions to guide the inference process of a data-driven method towards more plausible gene interactions that were already known, instead of solely reducing the complexity of the problem.

To address this, **we developed a novel approach called Fused Lasso Adaptive Prior (FLAP)**. FLAP is an extension of the FSSEM method, which is based on structural equation models (SEM).

While FSSEM is primarily data-driven, FLAP integrates prior knowledge about gene interactions to guide the GRN inference process and improve the results.

FSSEM, based on adaptive lasso linear regression, has proven to be among the best-performing state-of-the-art methods. This method integrates multi-omic data of gene expression and genotypes (SNPs) with genetic variant perturbations of eQTLs. Particularly noteworthy is FSSEM's capability to jointly infer two GRNs for different conditions (e.g., tumor vs healthy) and simultaneously optimize the identification of their differential network, obtained by subtracting the two inferred gene networks.

FLAP, however, extends FSSEM by incorporating prior knowledge of gene interactions into the inference process. This allows FLAP to not only rely on data but also leverage existing knowledge to improve the accuracy of the inferred gene regulatory networks.

We tested FLAP on synthetic data to assess its ability to account for the imperfections in prior knowledge, such as incorrect or missing edges. Subsequently, we applied FLAP to real data obtained from patients with lung cancer tissue and adjacent healthy tissue for control. Prior knowledge for these real data was obtained from various databases of gene interactions. We ran FLAP on these real datasets using different prior gene interactions to infer gene regulatory networks. Finally, we validated the resulting networks obtained with FLAP by assessing their biological plausibility using over-representation analysis (ORA) and through validation against existing literature.

Chapter 2: METHODS

2.1 Structural Equation Model

Structural Equation Model (SEM) is a linear model framework used by researchers to analyze complex relationships among variables. It encompasses various modeling techniques, including linear regression, multivariate regression, path analysis, confirmatory factor analysis, and structural regression. In fact, these models can be considered as particular cases of SEM, each with its own set of assumptions and applications.

SEM allows researchers to test hypotheses about how variables are related to each other and estimate causal relationships between them. Variables in SEM can be classified into two types: observed variables and latent variables. Observed variables are directly measured, while latent variables represent underlying constructs or concepts that cannot be directly observed and are inferred from the observed variables. For instance, the latent variable "intelligence" can be inferred from observed intelligence test scores.

Furthermore, variables in SEM can be categorized as either endogenous or exogenous. Endogenous variables are dependent variables and influenced by other variables in the system, while exogenous variables are independent variables not influenced by other variables in the system.

While SEM is most used in economics, sociology, and psychology, it has also found application in the field of biology to infer causal relationships.

In biology, a special case of SEM is often employed where all variables are observed, and there are no latent variables. This allows the SEM to not need a measurement equation model that infers latent variables from the observed ones, and it is simplified into having only a structural model of the form:

$$Y_i = BY_{-i} + FX_i + \epsilon_i$$

$$\begin{cases} Y_1 = BY_{-1} + FX_1 + \epsilon_1 \\ Y_2 = \beta Y_{-2} + FX_2 + \epsilon_2 \\ \quad \cdots \\ Y_p = BY_{-p} + FX_p + \epsilon_p \end{cases}$$

where

Y_i is the i-th endogenous variables

Y_{-i} represents all the endogenous variables except Y_i

X_i represents the exogenous variables assumed to influence Y_i

B is a matrix of regression coefficients for the endogenous variables Y_{-i}

F is a matrix of regression coefficients for the exogenous variables X_i

ϵ_i is the error term for the i-th equation, defined as a Gaussian vector of mean 0 and variance σ^2 and are independent and identically distributed (i.i.d.)

By including the exogenous variables in the model and estimating their effects using the matrix F, we are controlling for the effects of these external factors on the endogenous variables, thus the estimation of matrix B represents causal relationships and not a correlation.
Because the error terms are assumed to be uncorrelated, the SEM can be solved as a set of independent linear regressions, typically using maximum likelihood estimation (MLE).

2.1.1 GRN inference with SEM

In the context of inferring gene regulatory networks (GRNs), Structural Equation Model can be used to infer causal relationships between genes.
These causal relationships between genes are represented as a GRN by the regression coefficients in matrix B, while the causal relationships between perturbations and gene expression are represented by regression coefficients in matrix F. For instance, the coefficient $\beta_{i,j}$ indicates an edge from gene i to gene j, whereas $\beta_{j,i}$ indicates an edge in the opposite direction. The absolute value of the coefficient indicates the magnitude of the regulatory effect, while its sign indicates whether the regulation of the target gene expression is positive (promotion) or negative (inhibition).

In SEM-based GRN inference methods, the data consist of endogenous variables representing gene expression and exogenous variables representing gene expression perturbations.

Common types of perturbations include gene knockouts/knockdowns, which involve the deliberate manipulation of gene expression through techniques like CRISPR/Cas9 or RNA interference (RNAi); drug treatments, where specific drugs targeting gene expression are introduced to observe their effects on gene expression profiles; environmental perturbations, such as changes in temperature, pH, or exposure to specific substances; and expression Quantitative Trait Loci

(eQTLs), which are genetic variants (SNPs) in non-coding DNA regions affecting the expression of one or more genes (e.g., transcription factors binding sites, enhancers).

Among these perturbations, eQTLs are preferred for several reasons. Firstly, eQTLs are part of an individual's DNA sequence and are determined at birth, making them unaffected by environmental factors or other variables in the model. Secondly, they segregate in populations due to Mendelian inheritance, ensuring their independence from other variables within the model.

In SEM, these characteristics make eQTLs ideal as exogenous variables for inferring causality and were used in the works of (Cai et al., 2013) [41], (Logsdon and Mezey, 2010) [40] , (Liu et al., 2008) [39] and (Zhou et al. 2020) [5].

Under the assumption that each gene has at least one eQTL, the "Recovery" Theorem in (Logsdon and Mezey, 2010) guarantees that the network is uniquely identifieable. The presence of associations between eQTLs and genes (but not their effects) needs to be identified using methods like MatrixEQTL [56], which we will illustrate in the following section.

2.1.2 eQTL analysis with MatrixEQTL

The process of identifying SNPs significantly associated with the expression of genes is known as eQTL analysis. It enables the discovery of genetic factors (SNPs) involved in biological processes, diseases, and phenotypes, and helps in constructing causal networks. Causal variants often occur in noncoding DNA regions, where they can alter gene expression by affecting gene enhancers and binding sites for transcription factors.

The most common approach to eQTL analysis is to perform separate testing for each SNP-gene pair using linear or non-linear regression. Due to the size of the datasets, comprising millions of SNPs for the genotype and tens of thousands of gene transcripts, the number of SNP-gene pairs to test can reach into the billions, making the problem computationally intensive.

Applying non-linear methods has been shown to be too slow for even medium size datasets in the order of ten thousand SNP-gene pairs [57] [58] , thus it is often preferred the use of linear models.

MatrixEQTL is software developed to perform fast eQTL analysis using a simple linear regression model or the ANOVA model. This method allows for the identification of both cis-eQTLs and trans-eQTLs. Cis-eQTLs are genetic variants located close to the gene they regulate, often within the same chromosome (approximately 1 million base pairs) of their target gene. In contrast, trans-eQTLs are located farther away from the gene they regulate, sometimes even on

different chromosomes. Among these two types, cis-eQTLs perturb the gene expression of their target genes more significantly than trans-eQTLs and are often preferred as perturbations.

The MatrixEQTL simple linear regression model is defined as

$$\boldsymbol{Y} = \alpha + \beta_1 \boldsymbol{X} + \beta_2 \boldsymbol{C} + \epsilon$$

where

$\boldsymbol{Y}$ is the expression level of the gene

$\boldsymbol{X}$ is the SNP value of the genotype, encoded as {0, 1, 2} for the homozigous dominant (AA), heterozigous (aA and Aa) and homozigous recessive (aa) haplotypes.

α is the intercept of the model

β_1 is the regression coefficient reflecting the effect of the SNP on the gene expression

$\boldsymbol{C}$ are the covariates that may affect the gene expression (e.g. gender, age)

β_2 is the coefficient reflecting the effect of the covariates on gene expression

ϵ is the error term, a Gaussian random variable with zero mean and variance σ^2

Then, using this simple linear regression, the significance of the SNP $\boldsymbol{X}$ using test statistics can be calculated from the likelihood ratio (LR), t-statistic or the F-test and obtaining a p-value by testing for the hypothesis $\beta_1 \neq 0$ from the test statistics.

The Matrix EQTL ANOVA (ANalysis of VAriance) model tests the significance of a SNP-gene pair by assessing whether there is significant difference in gene expression levels among two different genotype groups. It can be seen as a linear regression

$$\boldsymbol{Y} = \alpha + \beta_1 \boldsymbol{x_1} + \beta_2 \boldsymbol{x_2} + \epsilon$$

where:

$\boldsymbol{Y}$ is the gene expression level of the gene

$\boldsymbol{x_1}$ is a dummy variable that is equal to 1 when the individual has the homozigous dominant genotype (AA) and 0 otherwise

x_2 is a dummy variable that is equal to 1 when the individual has teh heterozigous genotype (Aa) and 0 otherwise

α is the intercept of the model

β_1, β_2 are regression coefficients representing the association of the SNP with the gene expression level

ϵ is the error term, a Gaussian random variable with zero mean and variance σ^2

Here, the genotype variables are treated as categorical, assuming only two values: 1 and 0. Then it tests for a significant difference in the mean gene expression levels across these two different genotypic groups by testing for the null hypothesis that $\beta_i \neq 0, i = 1,2$, with an F-test from which it derives a p-value.

Both linear regression and ANOVA model compute the Benjamini-Hochberg method to control the false discovery rate (FDR) of SNP-gene associations.

2.2 Evolution of regression methods for GRN inference

As GRN inference methods evolved, those based on linear regression models also improved by incorporating different assumptions into their regression models. These assumptions were tailored to the biological characteristics of the GRN, leading to improved inference results. In this chapter, we explore the evolution of linear regression methods and their utilization within GRN inference (Fig. 1).

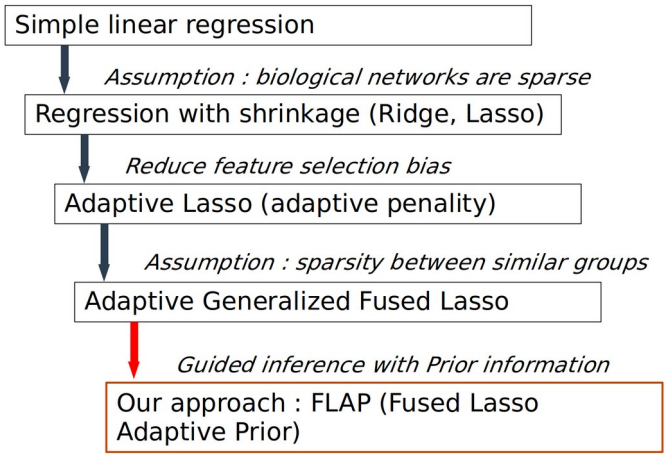

Fig. 1: Evolution of linear regression methods in GRN inference. Each successive method improves upon the previous one by incorporating a new concept to better define the model. FLAP aims to advance the state of the art by guiding the inference process through the integration of prior information.

Simple Linear Regression:

The simplest form of linear regression used in GRN inference is the simple linear regression model. It assumes a linear relationship between gene expression levels. Therefore, the network of interactions is derived by solving the linear regression between each gene g and the other (p – 1) genes, in the form:

$$\boldsymbol{Y_g} = \sum_{i=1, i\neq g}^{p} \beta_i \boldsymbol{Y_i} + \epsilon_g$$

where

Y_g is the expression level of the gene g

$Y_i = [Y_{1i}, Y_{2i}, \ldots, Y_{ni}]^T$ is the expression levels of gene i for n individuals.

β_i is the strength of the interaction between gene i and gene g, indicating an edge

ϵ_g is the error term representing the deviation from the observed value of Y_g

p is the total number of genes

The solution is given by minimizing the objective function:

$$\underset{\beta_i, i=1,\dots,p, i\neq g}{argmin} \|\boldsymbol{Y_g} - \sum_{i=1, i\neq g}^{p} \beta_i \boldsymbol{X_i}\|_2^2$$

By gathering all the β_{gi} coefficients in a $p x p$ matrix B we obtain our GRN.

The limitation of simple linear regression is that large coefficients can lead to overfitting. Overfitting occurs when the model learns the noise in the training data along with the underlying pattern, resulting in poor performance on new, unseen data. Regularized linear regression methods solve this problem by adding a penalty term to the simple linear regression objective function, penalizing large coefficients. This introduces sparsity in the results by shrinking some of the regression coefficients toward zero, as in Ridge regression, or exactly to zero, as in LASSO regression.

In the context of gene regulatory network inference, where there may be thousands of potential predictor variables (e.g. gene expression levels), sparsity is a desirable property. Sparsity allows the identification of a subset of regulator genes that have a significant impact on the expression of the target gene, leading to a more interpretable and precise model.

Ridge regression:

Ridge regression, performs linear regression by adding a L2 norm (also called Euclidean norm) penalty term to the simple linear regression objective function. This penalty term is proportional to the square of the magnitude of the coefficients, causing the estimated coefficients to shrink towards zero. This form of sparsity helps in reducing the impact of less important features (genes) while still retaining them in the model.

The Ridge regression objective function is expressed as:

$$\underset{\beta_i, i=1,\dots,p, i\neq g}{argmin} \left\| \boldsymbol{Y_g} - \sum_{i=1, i\neq g}^{p} \beta_i \boldsymbol{X_i} \right\|_2^2 + \lambda \sum_{i=1, i\neq g}^{p} \| \beta_i \|_2$$

where λ is the regularization parameter of the ridge penalty term that controls the amount of shrinkage to be applied to the coefficients.

LASSO regression:

LASSO (Least Absolute Shrinkage and Selection Operator) regression, like Ridge regression, adds an L1 norm penalty term to the simple linear regression objective function. The penalty term is proportional to the absolute value of the coefficients, enforcing exact sparsity by setting less relevant coefficients to zero. This property makes LASSO particularly useful in solving GRN inference as a feature selection problem, identifying gene regulators among many other genes.

The LASSO regression objective function is expressed as:

$$\underset{\beta_i, i=1,\dots,p, i\neq g}{argmin} \left\| \boldsymbol{Y_g} - \sum_{i=1, i\neq g}^{p} \beta_i \boldsymbol{X_i} \right\|_2^2 + \lambda \sum_{i=1, i\neq g}^{p} \| \beta_i \|_1$$

where λ is the regularization parameter of the penalty term that controls the amount of shrinkage to be applied to the coefficients.

Adaptive LASSO regression:

As mentioned earlier, LASSO regression improves upon simple linear regression by finding sparse solutions and shrinking many of the β_i coefficients to zero, while allowing the model to make predictions based on the few coefficients that are not zero. This reduces the prediction error of the model by decreasing the model complexity (i.e., the number of non-zero variables). However, as a side effect, it increases the bias of the estimation of β_i , known as the variance-bias tradeoff. The variance-bias tradeoff means that by reducing the variance, LASSO provides sparse solutions that are biased, so the variables that LASSO selects as meaningful can differ from the truly meaningful variables.

To address bias in the solution, the Adaptive LASSO [59] was developed as an "oracle" estimator. An estimator is considered oracle if it can correctly select the nonzero coefficients in a model with a probability converging to one (it identifies the right subset of true variables) and if the nonzero coefficients are asymptotically normally distributed (it achieves an optimal estimation rate).

This means tha given a set of p variables $\{\beta_1, \beta_2, ..., \beta_p\}$, if we consider two subsets

$$A = \{i : \beta_i \neq 0\} \rightarrow \textit{Truly significant variables}$$

$$\hat{A} = \{i : \hat{\beta}_i \neq 0\} \rightarrow \textit{Variables selected by the model}$$

an oracle estimator selects the truly significant variables with probability tending to one. Asymptotically, both subsets coincide.

The Adaptive LASSO objective function is

$$\underset{\beta_i, i=1,...,p, i \neq g}{argmin} \| \boldsymbol{Y_g} - \sum_{i=1, i \neq g}^{p} \beta_i \boldsymbol{X_i} \|_2^2 + \lambda \sum_{i=1, i \neq g}^{p} \hat{\boldsymbol{w}}_{\boldsymbol{i}} \| \beta_i \|_1$$

where λ is the regularization parameter and $\hat{\boldsymbol{w}}_{\boldsymbol{i}}$ is the adaptive weight that performs a different penalization for coefficient β_i to correct the bias in LASSO.

The adaptive weights are defined as

$$\boldsymbol{w_i} = \frac{1}{\left| \hat{\beta_i}^{initial} \right|^{\gamma}}$$

where $\hat{\beta_i}^{initial}$ is the initial estimate of the coefficients, usually obtained with Ridge regression and more rarely from Ordinary Least Squares (OLS) or LASSO regression and γ is a positive constant for adjustment of the adaptive weight (the authors suggest the possible values of 0.5, 1 and 2).

In the context of GRN inference, using Adaptive LASSO with its oracle property leads to better selection of truly significant coefficients representing gene interactions (network edges), thus reducing the number of false positives and false negatives. However, the exact impact on false positives and false negatives also depends on the specific dataset and the strength of the signals from the relevant predictors. Examples of works solving SEM using Adaptive LASSO to infer GRNs are (Cai et al., 2010) [41] and [40] (Logsdon and Mezey, 2010).

Adaptive Generalized Fused LASSO regression for joint modeling:

The assumption of sparsity characterizes Ridge, LASSO, and Adaptive LASSO regression, and holds for data that have samples from the same homogeneous group. However, in many real-world scenarios, data collected for analysis often exhibit structures that can be categorized into different strata, which are subsets of samples that share certain characteristics.

For instance, in epidemiological studies, data might be stratified based on factors such as age, gender, and ethnicity. In such scenarios, constructing independent sparse regression models for each stratum would not take advantage of the common structure. Conversely, constructing a single model for the entire dataset would mask the differences.

The work of [60] combined the adaptive LASSO and the generalized fused LASSO [61] into the adaptive generalized fused LASSO regression, a framework that enables the joint estimate of multiple sparse regression models for different strata.

The penalty is defined as follows:

$$penalty = \sum_{c=1}^{C} \{ \ \lambda_1 \sum_{j=1}^{p} w_j^{(1)} \ | \ \beta_{c,j} \ | \ \} \ + \ \lambda_2 \sum_{j=0}^{p} \sum_{c_1 > c_2} w_{c1,c2,j}^{(2)} \ | \ \beta_{c1,j} \ - \ \beta_{c2,j} \ |$$

where different strata are $(C_1, C_2, ..., C_n)$ and each stratum C can assume categorical values $c \in \{1, ..., C\}$, with $C \geq 1$ the total number of strata.

The first term enforces the lasso sparsity assumption of the adaptive LASSO with weights $w_j^{(1)} = | \hat{\beta}_j |^{-\gamma}$ and the second term is the fused penalty with weights $w_{c1,c2,j}^{(2)} = | \hat{\beta_{c1,j}} - \hat{\beta_{c2,j}} |^{-\gamma}$, represents the sparsity for coefficient β_j from two different strata c_1 and c_2. Unlike lasso sparsity, the fused penalty does not shrink the coefficients; instead, it encourages pairs of coefficients to have similar values by penalizing their absolute

difference. Parameters λ_1 and λ_2 govern the shrinkage for the lasso and fused lasso penalty terms.

The adaptive generalized fused lasso finds an application in the GRN inference method called Fused Sparse Structural Equation Model (FSSEM) [5] which can jointly infer GRNs for two different conditions (e.g. tumor vs healthy) and optimize the estimate for their difference network.

2.3 Fused Sparse Structural Equation Modeling (FSSEM)

The Fused Sparse Structural Equation Modeling (SEM) algorithm (Zhou et al., 2020) is designed to infer Gene Regulatory Networks (GRNs) across two different conditions simultaneously. It utilizes SEM with all observable variables of gene expression and gene perturbations, solving the inference problem by employing an adaptive generalized fused LASSO regression model.

The method utilizes gene expression levels under two different conditions (e.g., microarray or RNA-seq) along with gene perturbations (e.g., eQTLs or copy number variations). It particularly focuses on using cis-eQTLs as perturbations, leveraging the "Recovery" Theorem (Logsdon and Mezey, 2010). This theorem guarantees the identifiability of the network for both directed acyclic graphs (DAGs) and directed cyclic graphs (DCGs) when at least one eQTL is associated with each gene.

FSSEM uses the following SEM:

$$y_i^{(k)} = B^{(k)} y_i^{(k)} + F^{(k)} x_i^{(k)} + \mu_i^{(k)} + \epsilon_i^{(k)}$$

where

$k = 1,2$ are the two different conditions considered by the model

$i = 1, \ldots, n_k$ is the index of the considered gene for each condition $k = 1,2$

$B^{(k)} = [B^{(1)}, B^{(2)}]$ is a $n \times n$ matrix of p genes representing the unkown network structure under condition k

$F^{(k)}=[F^{(1)},F^{(2)}]$ is a $n\,x\,q$ matrix of p genes and q cis-eQTLs that captures the effect of of cis-eQTLs on gene expression.

$\mu_i^{(k)}$ is a $n\,x\,1$ vector that accounts for the model bias in the SEM

$\epsilon_i^{(k)}$ is a $n\,x\,1$ the vector the Gaussian noise with mean zero and variance σ^2

This SEM makes common assumptions found in GRN inference methods. First, it assumes no self-loops, meaning that the diagonal entries $B_{i,i}^{(k)}$ are set to 0. Secondly, it assumes independent Gaussian noise $\epsilon_i^{(k)}$.

Additionally, the SEM assumes that the q cis-eQTLs associated with each gene have been identified using eQTL analysis tools (e.g. MatrixEQTL), providing the SEM with the positions of the nonzero values of matrix $F^{(k)}$ but not their effects, which are estimated from the data. Lastly, the SEM assumes no prior knowledge about the inferred network $B^{(k)}$ and imposes no restrictions on the structure.

2.3.1 Joint inference of GRN in FSSEM

FSSEM defines the SEM as a negative log-likelihood function of the data

$$\begin{aligned} L(B,F,\mu,\sigma^2) &= -\log\prod_{k=1}^{2}\prod_{i=1}^{n_k} P(y_i^{(k)}|x_i^{(k)},\mu_i^{(k)},B^{(k)},F^{(k)}) \\ &= -\sum_{k=1}^{2}\frac{n_k}{2}\log\left|I-B^{(k)}\right|^2+\frac{(n_1+n_2)n}{2}\log(2\pi\sigma^2) \\ &\quad +\frac{1}{2\sigma^2}\sum_{k=1}^{2}\left\|(I-B^{(k)})Y^{(k)}-F^{(k)}X^{(k)}-\mu_i^{(k)}\right\|_F^2 \end{aligned} \quad (1)$$

where

$Y^{(k)}=[y_1^{(k)},\ldots,y_{n_k}^{(k)}]$ are the gene expression profiles

$X^{(k)}=[x_1^{(k)},\ldots,x_{n_k}^{(k)}]$ are the genotype profiles

n_k is the number of samples for condition k

The objective function is obtained by minimizing with respect to μ which yelds to $\hat{\mu}^{(k)}=(I-B^{(k)})\widetilde{Y}^{(k)}-F^{(k)}X^{(k)}$, where data are centered around the mean with

$\widetilde{Y}^{(k)} = Y^{(k)} - 1/n_k \sum_{i=1}^{n_k} y_i^{(k)} 1$, $\widetilde{X}^{(k)} = X^{(k)} - 1/n_k \sum_{i=1}^{n_k} x_i^{(k)} 1$ and integrating the adaptive generalized fused lasso penalty to obtain the following penalized negative log-likelihood function:

$$\begin{aligned} J(B,F) \;=\; & -\sum_{k=1}^{2} n_k \log\left|1 - B^{(k)}\right| \\ & + \frac{1}{2\hat{\sigma}^2} \sum_{k=1}^{2} \left\| \left(I - B^{(k)} \widetilde{Y}^{(k)} - F^{(k)} \widetilde{X}^{(k)} \right) \right\|_F^2 \\ & + \lambda \sum_{k=1}^{2} \left\| B^{(k)} \right\|_{1, w^{(k)}} + \rho \left\| B^{(2)} - B^{(1)} \right\|_{1, r} \end{aligned} \quad (2)$$

where

$\left\| B^{(k)} \right\|_{1, w^{(k)}} = \sum_i \sum_j w_{ij}^{(k)} \left| B_{ij}^{(k)} \right|$ is the adaptive lasso penalty term with weights $w_{ij}^{(k)} = 1/\left| \hat{B}_{ij}^{(k)} \right|$

$\left\| B^{(2)} - B^{(1)} \right\|_{1, r} = r_{ij} \left| B_{ij}^{(2)} - B_{ij}^{(1)} \right|$ is the adaptive generalized fused lasso penalty for two conditions with weight $r_{ij} = 1/\left| \hat{B}_{ij}^{(2)} - \hat{B}_{ij}^{(1)} \right|$

λ and ρ are the regularization parameters

The initial estimate of $B = [B^{(1)}, B^{(2)}]$ is obtained from the Ridge regression:

$$\{ \hat{B}, \hat{F} \} \;=\; \underset{\{B,F\}}{argmin} \sum_{k=1}^{2} \frac{1}{2} \left\| \left(I - B^{(k)} \right) \widetilde{Y}^{(k)} - F^{(k)} \widetilde{X}^{(k)} \right\|_F^2 + \lambda \left\| B^{(k)} \right\|_F^2 \quad (3)$$

from which is also obtained the estimate of σ^2 , $\hat{\sigma}^2$ for the previous objective function

$$\hat{\sigma}^2 \;=\; \frac{\sum_{k=1}^{2} \left\| \left(I - B^{(k)} \right) \widetilde{Y}^{(k)} - F^{(k)} \widetilde{X}^{(k)} \right\|_F^2}{\left(n_1 + n_2 \right) n} \quad (4)$$

This ridge regression problem can be decomposed into single ridge regressions, where each gene i is regressed against the other $(n-1)$ genes, and each gene i has a set of associated cis-eQTLs $\widetilde{X}_{S_q(i)}$ where $S_q(i)$ is the index of SNPs found to be associated with gene i by the eQTL analysis tool (e.g. MatrixEQTL)

$$\underset{\{B_{i,-i},F_{i,S_q(i)}\}}{argmin}\sum_{k=1}^{2}\frac{1}{2}\|\widetilde{Y}_i-B_{i,-i}^{(k)}\widetilde{Y}_{-i}^{(k)}-F_{i,S_q(i)}^{(k)}\widetilde{X}_{S_q(i)}^{(k)}\|_F^2+\lambda\|B_{i,-i}^{(k)}\|_F^2 \quad (5)$$

Minimizing the objective function in (5) w.r.t. $F_{i,S_q(i)}^{(k)}$ yelds to the closed form solution

$$F_{i,S_q(i)}^{(k)}=(\widetilde{Y}_i^{(k)}-\hat{B}_{i,-i}^{(k)}\widetilde{Y}_{-i}^{(k)})\widetilde{X}_{S_q(i)}^{(k)T}(\widetilde{X}_{S_q(i)}^{(k)}\widetilde{X}_{S_q(i)}^{(k)T})^{-1} \quad (6)$$

Substituting $F_{i,S_q(i)}^{(k)}$ into (5) and minimizing w.r.t. $B_{i,-i}^{(k)}$ gives the initial estimate of GRN $\hat{B}^{(k)}$

$$B_{i,-i}^{(k)}=\widetilde{Y}_i^{(k)}P_i^{(k)}\widetilde{Y}_{-i}^{(k)T}(\widetilde{Y}_{-i}^{(k)}P_i^{(k)}\widetilde{Y}_{-i}^{(k)T}+\lambda I)^{-1} \quad (7)$$

where $P_i^{(k)}=I-\widetilde{X}_{S_q(i)}^{(k)T}(\widetilde{X}_{S_q(i)}^{(k)}\widetilde{X}_{S_q(i)}^{(k)T})^{-1}\widetilde{X}_{S_q(i)}^{(k)}$

and when $B_{i,-i}^{(k)}$ is substituted in $F_{i,S_q(i)}^{(k)}$ gives the solution for the initial estimate $\hat{F}^{(k)}$

$$F_{i,S_q(i)}^{(k)}=\widetilde{Y}_i^{(k)}\Gamma_i^{(k)}\widetilde{X}_{S_q(i)}^{(k)T}(\widetilde{X}_{S_q(i)}^{(k)}\widetilde{X}_{S_q(i)}^{(k)T})^{-1} \quad (8)$$

where $\Gamma_i^{(k)}=I-P_i^{(k)}\widetilde{Y}_{-i}^{(k)T}(\widetilde{Y}_{-i}^{(k)}P_i^{(k)}\widetilde{Y}_{-i}^{(k)T}+\lambda I)^{-1}\widetilde{Y}_{-i}^{(k)}$

After $\hat{B}^{(k)}$ and $\hat{F}^{(k)}$ are estimated, the estimate of $\hat{\sigma}^2$ is given in (4).
The tuning parameter λ for the ridge regression is selected by 5-fold cross-validation.

Now the objective function $J(B,F)$ for the the SEM in (2) can be solved.
By Minimizing (2) w.r.t. $F^{(k)}$ yelds to $F_{i,S_q(i)}^{(k)}$ in (8). Substituting $F_{i,S_q(i)}^{(k)}$ (8) in $J(B,F)$ (2) gives an objective function for $J(B)$

$$J(B)=H(B)+\sum_{i=1}^{Ng}f_i(B_{i,-i}) \quad (9)$$

N_g number of genes

where

$$H(B) = -\sum_{k=1}^{2} \frac{n_k}{2} \log \left| I - B^{(k)} \right|^2 + \frac{1}{2\hat{\sigma}^2} \sum_{i=1}^{Ng} \sum_{k=1}^{2} \| \widetilde{Y}_i^{(k)} P_i^{(k)} - B_{i,-i}^{(k)} \widetilde{Y}_{-i}^{(k)} P_i^{(k)} \|_2^2 \quad (10)$$

and

$$f_i(B_{i,-i}) = \lambda \left(\| B_{i,-i}^{(1)} \|_{1,w^{(1)}} + \| B_{i,-i}^{(2)} \|_{1,w^{(2)}} \right) + \rho \| B_{i,-i}^{(1)} - B_{i,-i}^{(2)} \|_{1,r} \quad (11)$$

Because the function $J(B)$ is non-convex and non-smooth, the FSSEM algorithm minimizes it using the inertial version of the proximal alternating linearized minimization (iPALM) method [62]. This method employs block coordinate descent (BCD) optimization, which decomposes the objective function into blocks of variables. These blocks are optimized successively, with one block of variables optimized at a time while holding the others fixed. The hyper-parameters λ and ρ can be computed using grid search cross-validation (CV) or Bayesian Information Criterion (BIC).

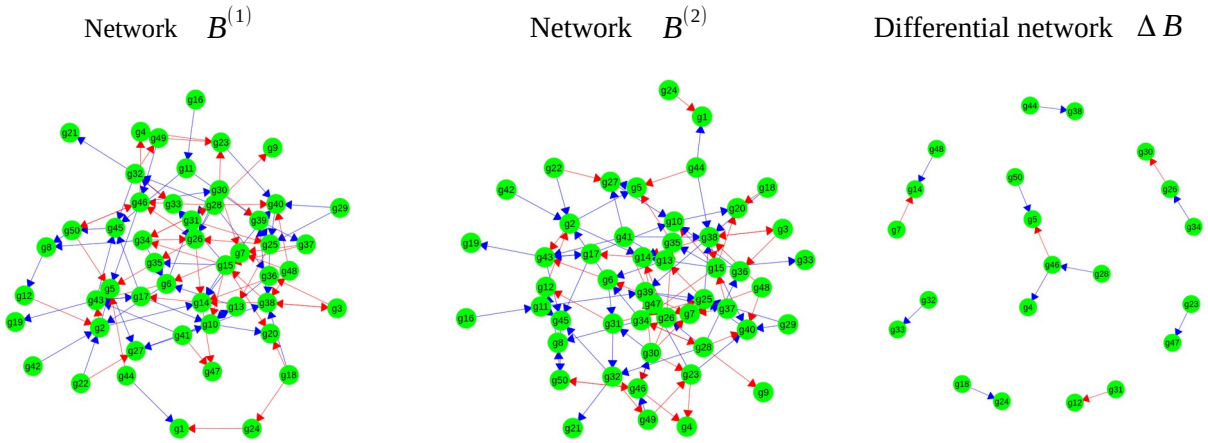

Fig. 2: Example of gene regulatory networks $B^{(k)}$ obtained with FSSEM from synthetic datasets. From left to right we have Network $B^{(1)}$, Network $B^{(2)}$ for the two cases k = 1 and k = 2, then we have the Differential Network $\Delta B = B^{(2)} - B^{(1)}$. *Blue* edges represent positive regulation (promotion), while *red* edges represent negative regulation (inhibition).

2.4 Fused Lasso Adaptive Prior (FLAP)

In our project, we developed the Fused LASSO Adaptive Prior (FLAP) method as an extension of the FSSEM method for GRN inference. FLAP utilizes prior knowledge of gene interactions by providing a flexible guide for the inference process, accounting for imperfections such as missing and incorrect interactions.

To achieve this, we modified the FSSEM method by directly incorporating prior knowledge into the ridge regression process, which forms the initial estimate. This initial estimate, obtained using ridge regression, is then used to create adaptive weights to guide the feature selection of relevant edges in the network. By integrating prior knowledge at this stage, the feature selection step is guided not only by the data but also by known gene interactions.

To integrate prior knowledge into ridge regression, we make use of penalty factors. Penalty factors are weights that determine the extent to which each coefficient is penalized during the model fitting process.

In ridge regression, the penalty term with penalty factors is given by:

$$penalty\ \ term\ =\ \lambda \sum_{j=1}^{Ng} \boldsymbol{p_j} \|\beta_j\|^2$$

where

λ is the regularization parameter

Ng is the number of genes in the dataset

p_j is the penalty factor associated with the j-th predictor.

β_j is the coefficient of the j-th predictor

By assigning penalty factors to individual coefficients, we can adjust the regularization applied to each variable in the model. This is particularly useful when certain variables are believed to be more or less important, or when there is prior knowledge suggesting that specific variables should be included or excluded from the regularization process.

For instance, assigning a higher penalty factor to certain variables implies that these variables will be more heavily penalized during the estimation process, effectively reducing their impact on the model predictions. On the other hand, assigning a penalty factor of zero to a variable means that

it will not be penalized at all, allowing it to be included in the model without any regularization. A common example is to not penalize demographic variables like sex and age in medical studies to include them in the final model.

Thus, in our FLAP method, integrating prior knowledge through penalty factors in ridge regression allows us to tailor the regularization to better account for the importance of different edges of the network. By improving the initial estimate we aim to improve the GRN inference process which heavily reliy on the definition of the penality term used during feature selection.

An important observation is that we designed FLAP to only integrate prior knowledge about the presence of interactions. This is because databases usually collect known interactions rather than their absence. Additionally, since every source of prior knowledge weights their edges differently, we cannot use them to scale our penalty factors. Therefore, we decided to set all penalty factors to the same chosen value. Next, we describe how we integrate the prior knowledge in FLAP.

The first step consists in creating an adjacency matrix A as a prior network of known edges of size $Ng\,x\,Ng$ where Ng is the number of genes in the dataset and $A_{i,j}=1$ where it exists a direct edge from gene i to gene j and $A_{i,j}=0$ otherwise.

The second step generates the matrix of penalty factors P from the prior network A and a choosen value for the penalty factors

$$P = (1 - A) + (penalty\,factor\,value * A)$$

where:

$(1 - A)$ is the opposite matrix of A, which creates the penalty factors = 1 of for the coefficients that will be fully penalized

$(penalty\,factor\,value * A)$ is the matrix that defines the penalty factors for the edges that are present in the prior network A and that will be penalized with the penalty factor value chosen by the user.

Reminding the solution of the initial estimate in the ridge regression step of FSSEM defined in formula (7)

$$B_{i,-i}^{(k)} = \widetilde{Y}_i^{(k)} P_i^{(k)} \widetilde{Y}_{-i}^{(k)T} \left(\widetilde{Y}_{-i}^{(k)} P_i^{(k)} \widetilde{Y}_{-i}^{(k)T} + \boldsymbol{\lambda I} \right)^{-1}$$

where the ridge regression had an uniform λI penalization and I is the identity matrix, we modified (7) to include penalty factors.

$$B_{i,-i}^{(k)} = \widetilde{Y}_i^{(k)} P_i^{(k)} \widetilde{Y}_{-i}^{(k)T} \left(\widetilde{Y}_{-i}^{(k)} P_i^{(k)} \widetilde{Y}_{-i}^{(k)T} + \lambda \boldsymbol{D_i^{(k)}} \right)^{-1}$$

where $D_i^{(k)} = diag(P_{i,-i}) = \{ p_{1,1}, p_{1,2}, \ldots, p_{1,i-1}, p_{1,i+1}, \ldots, p_{1,Ng} \}$ is the diagonal matrix obtained for gene i, without the column i since like in FSSEM we do not consider self-loops where gene i regulates itself.

Notice that we define $D_i^{(k)}$ with k = 1,2 the two different cases, allowing the user to define two separate prior networks. This feature can be useful in scenarios where there are different known interactions, such as paired data of tumor samples and healthy samples.

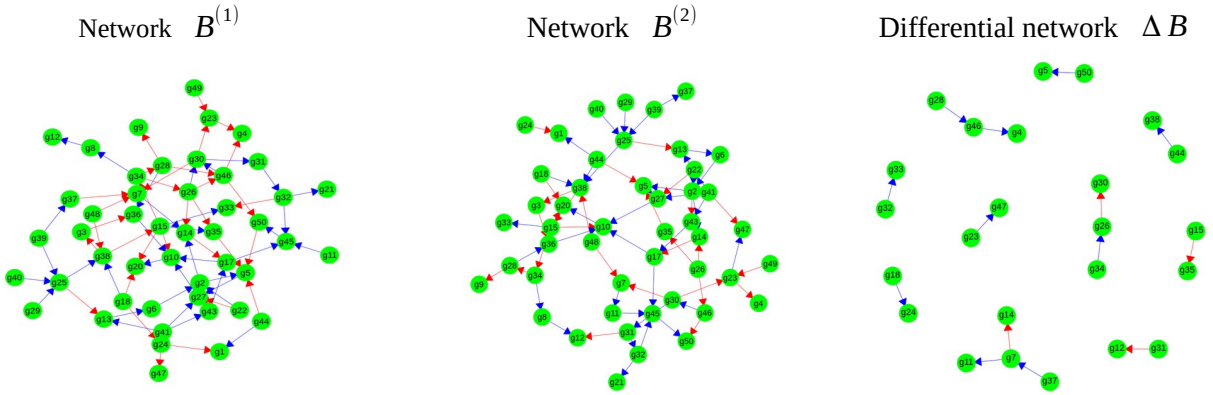

Fig. 3: Example of gene regulatory networks $B^{(k)}$ obtained with FLAP from synthetic datasets. From left to right we have Network $B^{(1)}$, Network $B^{(2)}$ for the two cases k = 1 and k = 2, then we have the Differential Network $\Delta B = B^{(2)} - B^{(1)}$. *Blue* edges represent positive regulation (promotion), while *red* edges represent negative regulation (inhibition).

2.4.1 Challenges in Integrating Prior Knowledge

Integrating prior knowledge into FLAP presented several challenges due to the possibility of incomplete or imperfect information. In particular, the prior knowledge about edges in the gene regulatory network could contain missing or erroneous information, which needed to be appropriately addressed.

The first challenge we encountered was determining the appropriate values for penalty factors to encode the prior knowledge about edges in the initial estimate of FLAP. This was crucial because penalty factor values range from 0, where the edge is always included in the result (no penalty), to less than 1, where the edge is favored to appear in the result (partial penalty). Thus, it was essential to find the optimal penalty factor values.

The second challenge was to determine whether integrating prior knowledge in the initial estimate effectively guided the inference process compared to integrating it as penalty factors in the second step of feature selection. We compared FLAP's performance in these two scenarios to evaluate which approach performed better when prior knowledge was imperfect.

step 1: penalty term in initial estimate (ridge regression)		step 2: penalty term in feature selection (adaptive generalized fused lasso)
$\lambda \sum_{k=1}^{2} \sum_{i} \sum_{j} \boldsymbol{p_{ij}^{(k)}} \lVert \beta_{ij} \rVert^2$	VS	$\lambda \sum_{k=1}^{2} \sum_{i} \sum_{j} \boldsymbol{w_{ij}^{(k)}} * \boldsymbol{p_{ij}^{(k)}} \lvert B_{ij}^{(k)} \rvert + \rho \lVert B^{(2)} - B^{(1)} \rVert_{1,r}$

We obtained the adaptive weights combined with the penalty factors $w_{ij}^{(k)} * p_{ij}^{(k)}$ by multiplyaing the adaptive weights matrix $W^{(k)} = 1/\lvert B^{(k)} \rvert$ with the penalty factors matrix P, which can be expressed as $(W^{(k)})^T P^{(k)}$.

In the next chapter we address these challenges, testing our FLAP method on synthetic datasets and evaluated the performance by comparing the results with the gold standard network that generated the synthetic data. This allowed us to assess how well FLAP performed in the presence of incomplete or imperfect prior knowledge.

Chapter 3: Testing and Validation of FLAP on Synthetic and Real Data

3.1 Synthetic Data Simulations

In this section we present the tests with FLAP on syntetic data to solve the two challenges in integrating prior knowledge, we evaluate FLAP robustness to noise and compare FLAP with FSSEM and BDFSEM, the other two state-of-the-art methods for joint inference of GRNs.

3.1.1 Generation of Synthetic Dataset

Following the setup of SML [41] and FSSEM [5], we generated data for directed acyclic graphs (DAGs). However, we chose not to simulate synthetic directed cyclic graphs (DCGs) because this setup generates the network structure edges by adding random edges without controlling for the presence or number of cycles. Therefore, we limited our study to the DAG case.

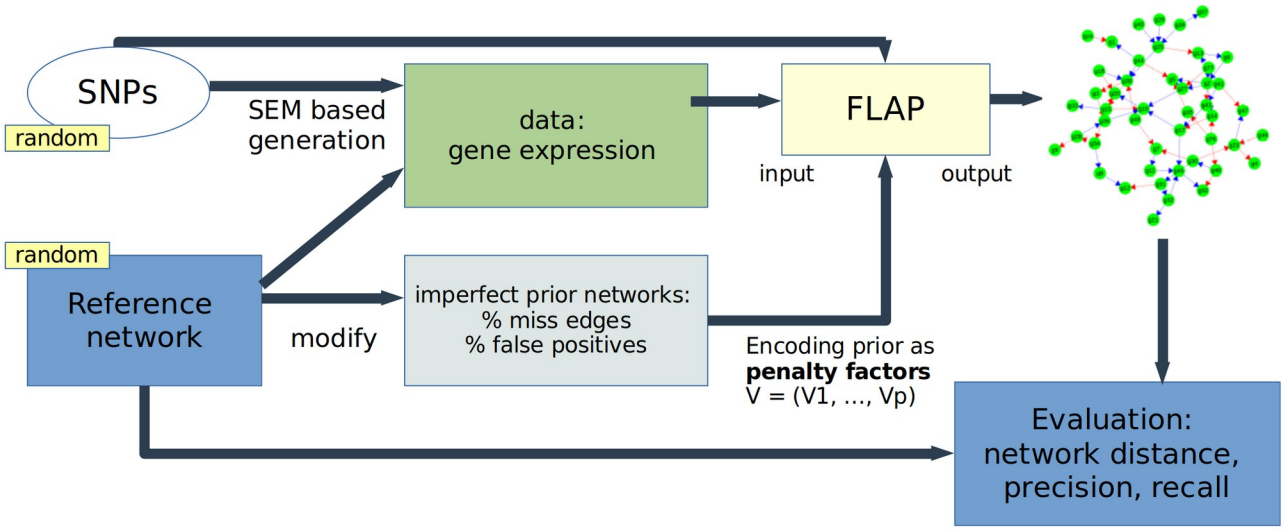

Fig. 4: Scheme of synthetic dataset generation. Gene expression data are generated from a random reference network and random SNPs using the SEM equation. Imperfect prior networks are generated by modifying the reference network that generated the data. FLAP takes in input gene experssion, SNPs, SNPs-gene associations and prior networks encoded as penalty factors. The resulting networks are evaluated by comparison with the reference network.

First, the two reference gene networks $B^{(k)}$ for $k=1,2$ are constructed by randomly generating their structure in the form of adjacency matrices $A^{(k)}_{p\times p}$ for $k=1,2$ and p is the number of genes. In these matrices, a value of 1 indicates the presence of an edge between two genes, and a value of 0 indicates the absence of an edge.

The structure is built starting from the adjacency matrix $A^{(1)}$. Given the number of genes p and the desired average number of edges s, the total number of edges for $A^{(1)}$ is generated using a random binomial distribution defined as

$$total\ \ number\ of\ edges\ \sim\ \mathrm{Bin}\left(d=p\times p,\ \frac{s}{(p-1)}\ \right)$$

where

$d=pxp$ is the number of possible edges in the network

$\frac{s}{(p-1)}$ is the probability of success, meaning the probability of a gene to have an edge with any of the other $(p-1)$ genes

The adjacency matrix $A^{(1)}$ is initially filled with zeros and a new edge is iteratively added until reaching the previously generated total number of edges. Each new edge is represented as an additional 1 in one of the $d=p\times p$ possible indexes of the adjacency matrix $A^{(1)}$ and this position is generated using a uniform random distribution $U(1,d)$.

Then, the adjacency matrix $A^{(2)}$ for condition 2 is obtained by randomly changing a percentage df of the 0 and 1 entries of $A^{(1)}$. By default df is set to 10%.

From the adjacency matrices $A^{(1)}$ and $A^{(2)}$ representing the GRN structure, the weights of the network edges $B^{(1)}$ and $B^{(2)}$ are generated as follows: for any entry $A_{ij}=1$, a value B_{ij} is assigned from a random variable uniformly distributed over the interval [0.5, 1] or [-1, -0.5].

The genotypes of k SNPs were simulated using the R package “qtl”, selecting second filial generation cross (F2 cross), with values of 1 and 3 representing the dominant and recessive homozygous genotypes, and 2 representing the heterozygous genotype.

Subsequently, two genotype data matrices, $X^{(1)}$ and $X^{(2)}$, were generated for conditions 1 and 2, respectively, by randomly sampling {1, 2, 3} with corresponding probabilities {0.25, 0.5, 0.25}.

The regulatory effects of the corresponding eQTLs were assumed to be 1 (and 0 otherwise) and were stored as matrices $F^{(1)}$ and $F^{(2)}$ of size $p \times k$. For the synthetic dataset we assumed that each gene has the same number of eQTLs $\frac{k}{p}$.

The error terms $E^{(1)}$ and $E^{(2)}$ of the SEM simulate the noise in the generated data that can arise from various sources such as measurement errors, environmental factors, or other unexplained variability. Because we are in the context of linear regression the error terms are assumed to be normally distributed.

Each error term $E^{(k)}$ for $k = 1,2$ is obtained from a multivariate normal distribution

$$E^{(k)}_{n \times p} \sim \boldsymbol{N}\left(\mu = 0,\ \Sigma = \sigma^2 \mathbf{I_p}\right)$$

where

- n is the number of samples for the generated data
- p is the number of genes
- $\mu = 0$ is the mean equal to zero
- Σ is the covariance matrix
- I_p is the identity matrix of size p
- σ^2 is the noise variance, it quantifies the spread or dispersion of the random error term. A high value indicates more variability in the noise, while a lower value indicates that the noise is more tightly clustered around the mean, which is zero.

At last, the gene expression level $Y^{(1)}$ and $Y^{(2)}$ are calculated using the formula of the SEM

$$Y^{(k)} = \left(I - B^{(k)}\right)^{-1}\left(F^{(k)} X^{(k)} + E^{(k)}\right)$$

where $k = 1,2$ are condition 1 and condition 2.

In summary a synthetic dataset is generated by setting the number of samples n, the number of genes p, the number of SNPs k, the average number of edges per gene s, the noise variance σ^2 and the percentage of differential edges among the two networks df.

3.1.2 Generation of Synthetic Prior Network

The prior networks we generated utilize the gold standard gene regulatory networks used to generate the data, represented by the weighted edge matrices $B^{(1)}$ and $B^{(2)}$ for case and control, respectively.

From each gold standard network, we converted it into an adjacency matrix of zeros and ones, representing a perfect prior network.

Subsequently, we generated imperfect prior networks by modifying the perfect prior network. The prior network with correct and missing edges was generated by randomly converting a chosen percentage of the adjacency matrix values from 1 to 0. This type of prior network will be used in sections 3.1.4 and 3.1.5.

We then generated two types of prior networks with correct and additional wrong edges. The first type (type1) was created by adding to the perfect prior network a number of random edges equal to a percentage of the perfect prior edges. For example, given a perfect prior network of 100 edges, an imperfect prior with additionally 20% of wrong edges would have 20 more wrong edges. This type of prior network will be used in section 3.1.4. The second type (type2) is generated by adding to the perfect prior network a number of random edges equal to a percentage of perfect prior edges while also removing the same number of correct edges. This generates a prior with a constant total number of edges. At a percentage of 100% additional wrong edges, the prior network will be comprised solely of wrong edges, matching the count of the edges in the perfect prior. Percentages above 100% remove all the correct edges but add more wrong edges than were originally present in the perfect prior. This type of prior will be used in section 3.1.5.

3.1.3 Classification and performance metrics

In network analysis, edges can be classified by comparing an evaluated network to a reference network that represents the true set of interactions. These edges are categorized into four classifications: true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN).

True positives (TP) are edges that are present in both the evaluated network and the reference network, representing correct interactions.

False positives (FP) are edges that are present in the evaluated network but not in the reference network, indicating interactions that do not actually exist.

True negatives (TN) are edges that are absent in both the evaluated network and the reference network, representing correctly identified non-interactions.

False negatives (FN) are edges that are absent in the evaluated network but present in the reference network, representing missed interactions.

In this thesis, these classifications are used to describe imperfect prior networks in relation to the reference perfect prior network. Imperfect prior networks have missing edges that are false negatives, wrong edges that are false positives, and correct interactions and non-interactions that are true positives and true negatives, respectively.

In addition to using classification metrics to describe prior networks, we also utilize them to evaluate the accuracy of inferred networks. When comparing an inferred network (the evaluated network) to a gold standard network that generated the synthetic data (the reference network), we use classification metrics (TP, FP, TN, FN) to compute performance metrics. These performance metrics, which include precision, recall, and accuracy, provide an assessment of the network's accuracy.

Precision measures the proportion of correctly predicted interactions among the total positive predicted interactions. It shows how often the model is correct when predicting the positive class.

$$Precision = \frac{TP}{TP+FP}$$

Recall measures the proportion of correctly predicted interactions among the total true interactions in the gold standard network.

$$Recall = \frac{TP}{TP+FN}$$

Accuracy measures the proportion of correctly predicted interactions (both positive and negative) among all interactions.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

3.1.4 Challenge 1: Calibration of penalty factors

To determine the appropriate values for the penalty factors to use in the ridge regression for the initial estimate, we tested FLAP on synthetic datasets (see section 3.1.1) and their associated prior networks with different levels of imperfect information.

For each dataset, we generated 20 replicates, each comprising of $n=50$ samples and $p=50$ genes. Each gene was associated with 3 eQTLs, totaling $k=150$ eQTLs considered. The underlying reference network was generated to have an average number of edges per gene of $s=1.5$ and a noise variance $\sigma^2=0.25$.

For each dataset, we generated the following prior networks: one perfect prior network identical to the structure of the reference network, three prior networks with 75%, 50%, and 25% of missing edges, and six prior networks with 25%, 50%, 75%, 100%, 150%, and 200% of incorrect edges added to the perfect prior network.

The penalty factor values ranged from 0, where the prior edges are not penalized, to 1, where the prior edges are fully penalized. Additionally, we considered intermediate values of 0.25, 0.5, and 0.75, where the prior edges are partially penalized. Subsequently, we tested FLAP with each combination of prior network and penalty factors.

To evaluate the performances of the resulting networks, we compared them with their gold standard counterparts using the Edge Difference Distance (EDD) method from the R package "NetworkDistance" [63] . The resulting network distances were averaged over the 20 replicates for each dataset.

This method measures the distance between two networks by computing the Frobenius norm of their difference, considering the weights of the edges of the networks.

We calculated the difference between the networks obtained from the results

$$\Delta B_{inferred} = B2_{inferred} - B1_{inferred}$$

and their corresponding gold standard networks

$$\Delta B_{gold\ standard} = B2_{gold\ standard} - B1_{gold\ standard}$$

for each combination of prior and penalty factor values.

$$distance_{EDD}(\Delta B_{gold\ standard}, \Delta B_{inferred}) = \|\Delta B_{gold\ standard} - \Delta B_{inferred}\|_F$$

The resulting network distances are illustrated in the following 3D barplot.

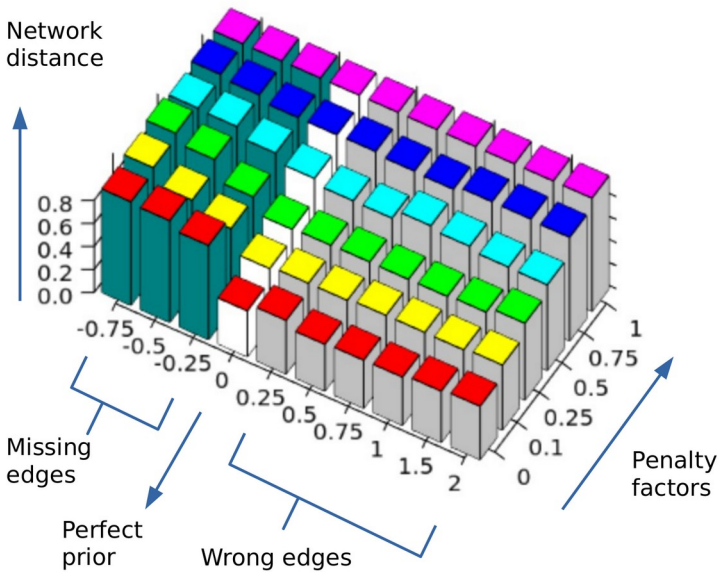

Fig. 5: Network distance evaluated by comparing the inferred differential network with the gold standard differential network. The smaller the distance the better the result. The plot shows the performance of FLAP at the varying of penalty factors value for the perfect prior and imperfect prior with missing or additional wrong edges.

From the resulting network distances, we found that the network distance was minimal when the edges of the prior network were encoded as zeros. Additionally, we observed that the distance for the same prior network increased as the penalty factor value increased.

This was expected because, as the penalty factor value increased, the impact of the prior edges became smaller, and the resulting network became more dependent on the data. Eventually, with penalty factors equal to 1, the network relied solely on the data.

Based on these observations, we conclude that the most effective way to integrate prior network information in the ridge regression step is to set the penalty factors to 0, thus removing their penalization in the initial estimation step.

3.1.5 Challenge 2: Evaluating the optimal step to integrate prior knowledge

To assess whether integrating prior knowledge as penalty factors into the initial estimate of the ridge regression step was more effective than using penalty factors in the second step of the feature selection, we compared the performance of the two approaches using the following synthetic datasets (see section 3.1.1) and synthetic prior networks.

The synthetic dataset we used is defined like the previous case, comprising of $n=50$ samples, $p=50$ genes, with 3 eQTLs associated with each gene, resulting in a total of $k=150$ eQTLs. The expected average number of edges per gene was set at $s=1.5$ and the noise variance at $\sigma^2=0.25$.

We created two types of prior networks: a prior network with correct and missing edges, and a prior network with correct and additional wrong edges (type2) (see section 3.1.2).

Prior networks with missing edges:

We created five types of priors with missing edges. In each new prior, the proportion of correct edges (True Positives) decreased by 25%, while the missing edges (False Negatives) increased by 25%. These priors ranged from a 100% correct prior with 0% missing edges (perfect prior) to a 0% correct prior with 100% missing edges (no prior).

Prior networks with additional wrong edges:

Similarly, we created seven types of priors with false positive edges. In each new prior, the proportion of correct edges (True Positives) decreased by 25%, while the false positive edges (False Positives) increased by 25%. These priors ranged from a 100% correct prior with 0% false positive edges (perfect prior) to a 0% correct prior with 200% false positive edges (no prior).

We then evaluated the performance using precision and accuracy metrics to compare the resulting differential network with the gold standard differential network.

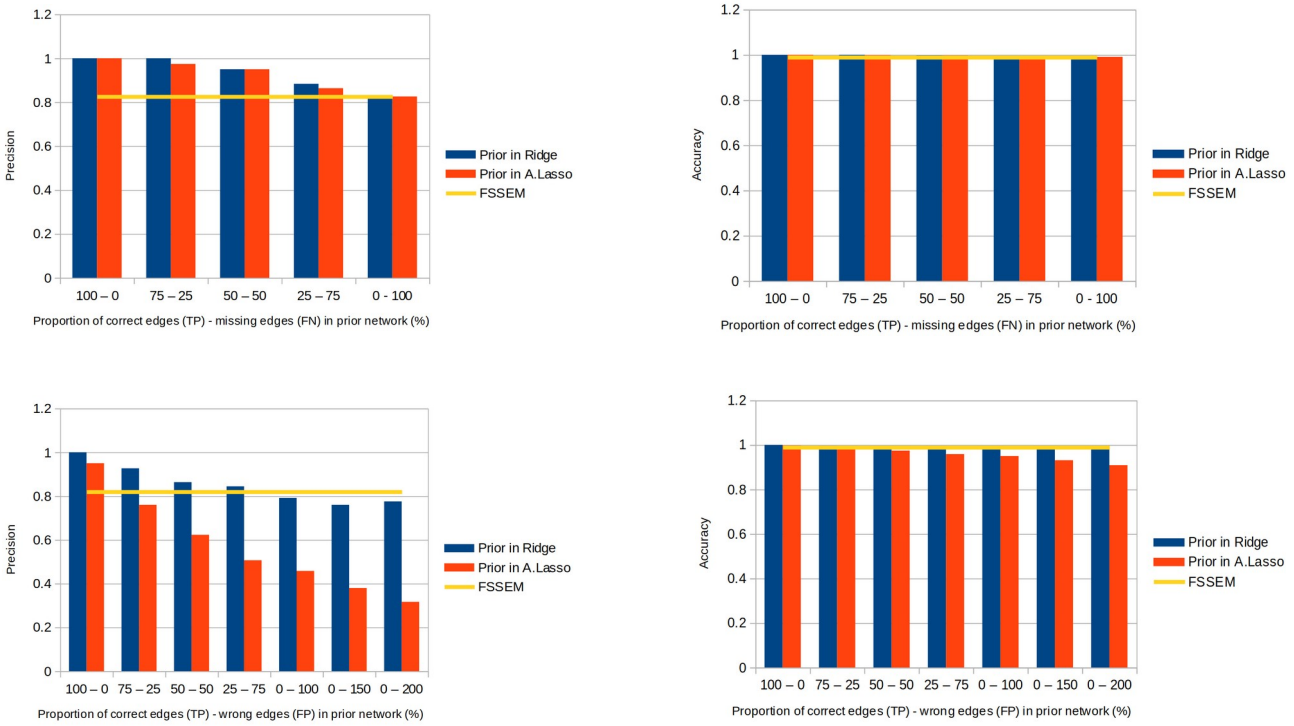

Fig. 6: Precision and accuracy values for the integration of prior knowledge in the initial estimate of the network using ridge regression (*blue*) and in the feature selection step using adaptive generalized fused lasso (*red*). The performance of FSSEM is shown in *yellow* for comparison. The plots show that integrating prior knowledge in the initial estimate with ridge regression is more robust in the presence of false positives in the prior network, resulting in better precision.

The results show that the presence of false positive edges in the prior network noticeably affects precision and accuracy. Integrating the prior network in the first step of the initial estimate yields better results compared to integrating it in the second step of feature selection. The performance deteriorates more rapidly with an increased proportion of false positive edges.

For instance, the values for precision fall below the level of not using any prior information when the prior network is integrated in the first step of the initial estimate, specifically at 0% correct edges and 100% false positives. Similarly, when using prior knowledge in the second step of feature selection, precision falls below the level of not using any prior information at 75% correct prior and 25% false positives.

This result demonstrates that integrating prior knowledge in the initial estimate of the network works as a flexible guide for the inference process, providing our FLAP method with robustness in case of incorrect priors, both for missing edges and false positives.

3.1.6 Robustness to noise

In this test, we evaluated the robustness of our FLAP gene network inference method to noise. We tested the performance of the method using synthetic datasets generated with various levels of noise variance: $\sigma^2=0.01$ (low), $\sigma^2=0.1$ (medium), and $\sigma^2=0.25$ (high).

Three datasets were generated, each with $n=50$ samples, $p=50$ genes, with 3 eQTLs associated with each gene, making a total of $k=150$ eQTLs and an expected average number of edges per gene $s=1.5$. The datasets were created using the same seed to ensure that they differ only in their noise levels.

For each dataset, we ran FLAP twice with two different priors. The first prior was the perfect prior, while the second was a random prior obtained by bootstrapping the perfect prior 100 times. The bootstrapping of the prior network was performed using the R package “igraph” [64] . We utilized the function “rewire” with the option “keeping_degseq(niter=100)”, which executes 100 iterations of the rewiring algorithm. In each iteration, the rewiring algorithm selects two arbitrary edges ((a,b) and (c,d)) and substitutes them with (a,d) and (c,b) if they do not already exist in the graph.

Once again, we evaluated the performances of the resulting networks, by comparing the inferred differential networks with the gold standard differential network using the Edge Difference Distance (EDD) method from the R package "NetworkDistance".

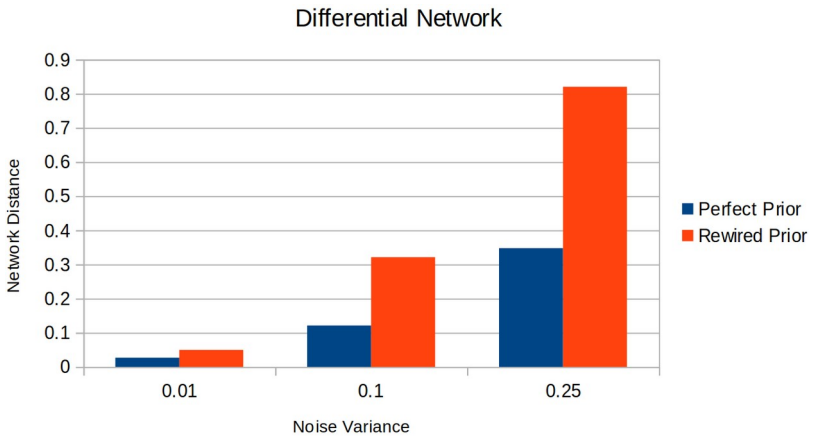

Fig. 7: Network distance for the differential network using data with varying level of noise variance $\sigma^2 = 0.01, 0.1, 0.25$ and comparing the use of perfect prior network (*blue*) and a randomly rewired prior network (*red*) with the same connectivity. The plot shows that the performance of FLAP do not depend on the connectivity of the network used but on the correct prior information.

The results show that at low noise variance ($\sigma^2 = 0.01$), both the perfect prior and rewired prior resulted in small network distances, with the rewired prior showing slightly higher (worse) values. As noise variance increased to medium ($\sigma^2 = 0.1$) and high ($\sigma^2 = 0.25$) levels, the difference in network distance between the perfect prior and rewired prior became more pronounced. In both cases, the perfect prior consistently outperformed the rewired prior, resulting in significantly lower network distances.

These results suggest that the FLAP method is robust to noise, and its performance is attributed to the information provided by the prior rather than the properties of the prior network used. This conclusion is supported by the fact that the rewired prior, while having the same properties as the perfect prior, had different edges.

3.1.7 Comparing FLAP with FSSEM and BDFSEM

We compared the performance of our method, FLAP, with two other methods: FSSEM and its Bayesian version, BDFSEM.

For this comparison, we used the same synthetic datasets and gold standard networks as in the previous sections. FLAP was tested with both data and a perfect prior network, while FSSEM and BDFSEM relied solely on data.

Additionally, we conducted tests on synthetic data with varying numbers of samples. FSSEM and BDFSEM showed improved performance with an increased number of samples, as reported in their original papers. Therefore, we tested FLAP in the "small n large p" scenario, where the number of samples is smaller than the number of features, as well as when the number of samples was equal to or greater than the number of features, to determine if our method had the potential to outperform them under these conditions.

The synthetic dataset consisted of $p = 50$ genes, $k = 150$ SNPs, and had noise variance $\sigma^2 = 0.25$. Sample sizes n tested were 40, 50, 80, 100, and 150.

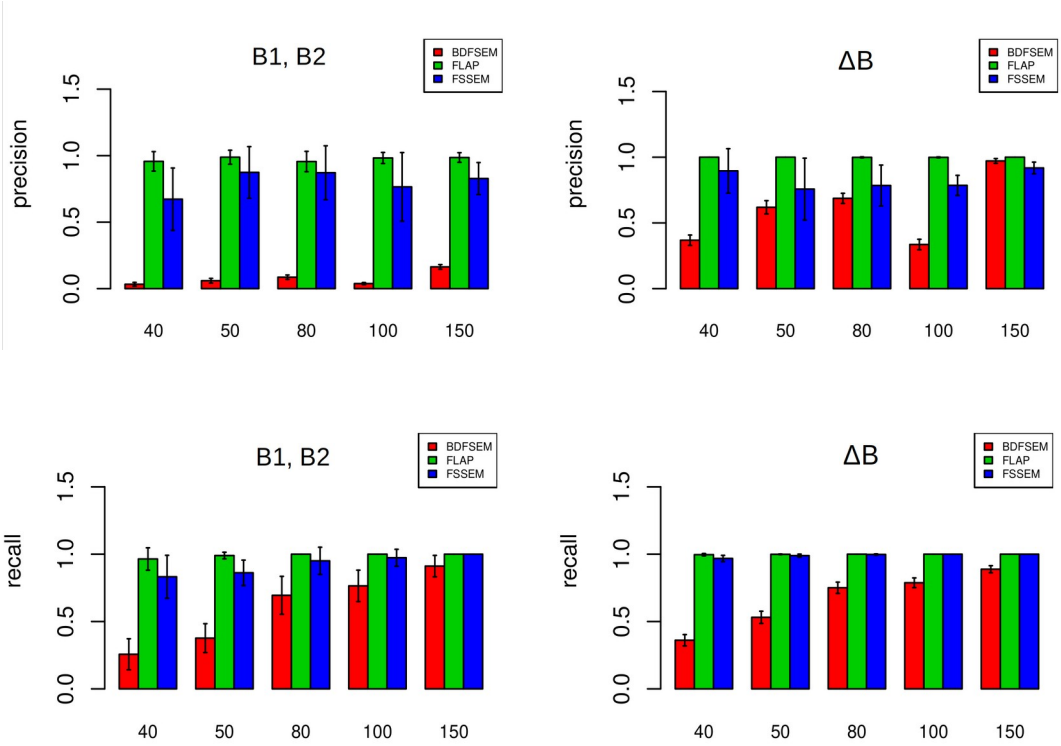

Fig. 8: Performance comparison of FLAP (green), FSSEM (blue), and BDFSEM (red). The x-axis represents the number of samples in the datasets, ranging from 40 to 150, while the y-axis represents the values for precision and recall for the average of the single networks of case and control (B1,B2) and their differential network (ΔB = B2 - B1).

From the resulting plot we conclude that FLAP with the perfect prior achieves better performances compared to FSSEM and BDFSEM. BDFSEM requires a higher number of samples to achieve comparable performance to FLAP and FSSEM.

3.2 Real data analysis

After testing our method on synthetic data, we proceeded with our research to assess its capacity in inferring gene networks using real biological data. This step allowed us to evaluate its practical applicability and to validate whether the inferred gene networks can effectively identify relevant biological processes.

For our study, we utilized data sourced from the Gene Expression Omnibus (GEO) database under the accession number GSE33356 [65] . This dataset provides genome-wide screening results of genotype profiles and microarray gene expression profiles from non-smoking female lung cancer patients in Taiwan. It is comprised of two subseries datasets: one for gene expression data and one for genotype data. Details are reported in the following table:

Data Type	Platform	SubSeries	Number of Probes	Numper of Paired Samples
Gene Expression	GPL570: [HG-U133_plus_2] Affymetrix Human Genome U133 Plus 2.0 Array	GSE19804	54,675	60 pairs of cancer and normal lung tissue samples
Genotype (SNP)	GPL6801: [GenomeWideSNP_6] Affymetrix Genome-Wide Human SNP 6.0 Array	GSE33355	906,551	61 pairs of cancer and normal lung tissue samples

Table 1: SubSeries composing the dataset GSE33356. SubSeries GSE19804 contains the gene expression profiles, while SubSeries GSE33355 contains the genotype (SNP) profiles.

For our analysis, we considered only the subjects that had paired samples in both the gene expression and genotype datasets. This resulted in a total of 42 patients, each with 4 profiles: a pair of gene expression profiles (one for cancer tissue and one for normal tissue) and a pair of genotype profiles (one for cancer tissue and one for normal tissue).

The raw microarray gene expression data of 54,675 probes were normalized using the R package affy [66] with custom Brainarray CDF version 25 (released on Jan 5, 2021) [67] . The normalization method applied was the robust multi-array average (RMA) method [68] to derive gene expression levels.

In total, we acquired gene expression levels for 20,422 genes along with their Entrez IDs. Subsequently, we retrieved gene annotations from the Ensembl database using the R package biomaRt [69] , querying the Entrez IDs on the Genome Reference Consortium Human Build 37

(hg19). Following filtration for protein-coding genes, we were left with 16,925 genes and their respective locations.

The genotypes of 906,551 SNP probes were converted into SNP identifiers utilizing the annotation file “GenomeWideSNP_6.na35_annot.csv.zip” obtained from the Affymetrix official site. This annotation file also provided the positions of the SNPs on the chromosomes.
The raw genotype data were converted from unphased haplotypes into numerical values using the mapping AA: 0 , AB: 1, BB: 2. Any missing genotypes were imputed using the R package Synebreed (version 0.12-14) [70] using the Beagle imputation method [71] .
SNPs were filtered to keep those with minor allele frequency (MAF) above 0.05 [72] , aiming to enhance the statistical power of the SNPs potentially associated with gene expression level.

Next, we utilized the R package MatrixEQTL to detect cis-eQTLs within a 1 M base pair (bps) range from the open reading frame (ORF) of the gene. MatrixEQTL was run as linear regression model for each gene-SNP pair, including covariates for patient sex and tissue type (tumor or normal). From the resulting cis-eQTLs, we selected those with p-value < 1e^-4 and FDR < 0.05. Additionally, we filtered those with SNPs having a MAF > 0.05 in both tumor and control samples, ensuring genetic variability within each condition.
This process yielded a total of 3002 cis-eQTLs, involving 1100 genes and 1848 SNPs.
Given the algorithm's primary objective of identifying differential gene networks between tumor and normal tissues, we focused on differentially expressed genes likely to be biologically relevant in distinguishing between the two conditions. Consequently, we conducted a differential gene expression analysis using the R package limma [73] , selecting differential genes with adjusted p-values < 0.01.
Subsequently, filtering our dataset for differential genes that have at least one cis-eQTL associated, we obtained a dataset of 289 genes and 463 cis-eQTLs.

After generating the dataset, we used the list of genes within it to extract the necessary networks of prior knowledge required to guide the gene network inference process. The prior knowledge was obtained from the databases of GIANT, TissueNexus, STRING, hTFtarget and Harmonizome 3.0.

GIANT database:

The Genome-wide Integrated Analysis of Networks in Tissues (GIANT) database provides tissue-specific gene networks for a collection of 144 human tissues. These networks feature weighted undirected edges that represent functional associations among genes. The edges connect genes involved in the same pathway or biological process, with the edge weights indicating the confidence of the connection in terms of probability (e.g., an edge weight of 0.8 means there is an 80% confidence in the existence of the edge).

Since our dataset comes from lung cancer patients, we retrieved the network for lung tissue (file: “lung_top.tsv”) from the GIANT download section. This network initially comprised 25,825 genes and 59,798,192 edges. We then filtered this network for the 289 genes in our dataset, resulting in a network of 232 genes and 3,037 edges. These undirected edges were split into 6,074 directed edges.

To generate a high-confidence prior network, we subset this network to include only edges with weights above the threshold of 0.2, removing low-confidence edges. This filtering resulted in a network comprised of 119 genes and 416 undirected edges, which were split into 832 directed edges.

TissueNexus database:

The TissueNexus database comprises 49 tissue-specific human gene networks for a collection of 49 human tissues. These networks feature weighted undirected edges that represent functional associations among genes, where edges connect genes involved in the same pathway or biological process. The edge weights indicate the confidence of the connection in terms of probability.

We obtained the gene network specific to lung tissue (file: “lung.zip”) from the download section of the TissueNexus site (https://www.diseaselinks.com/TissueNexus/download.php). This network initially comprised 16,889 genes and 5,171,074 edges. We then filtered this network for the 289 genes in our dataset, resulting in a network of 271 genes and 1,704 edges. These undirected edges were split into 3,480 directed edges.

To generate a high-confidence prior network, we subset this network to include only edges with weights above the threshold of 0.6, removing low-confidence edges. This filtering resulted in a network comprised of 242 genes and 653 undirected edges, which were split into 1,306 directed edges.

STRING database:

The Search Tool for the Retrieval of Interacting Genes/Proteins (STRING) is a database that enables users to construct protein-protein interaction networks. It integrates information from various sources to build network interactions and calculates confidence scores for the identified protein-protein interactions.

In our study, we utilized STRING to query our dataset of 289 genes using their Entrez IDs, specifying the organism as "human." We filtered the resulting network to include only experimental sources and a confidence level above 0.150 (low) to ensure an adequate number of gene interactions. This filtering process yielded a network of 181 genes and 249 undirected edges. Subsequently, we converted these undirected edges into directed ones, resulting in a total of 498 directed edges.

hTFtarget database:

The hTFtarget is a comprehensive database of human transcription factors (TFs) and their targets, constructed by integrating resources from ChIP-seq experiments of 659 TFs, high-confidence binding sites of 699 TFs, and epigenetic modification information.

From the hTFtarget site download section (http://bioinfo.life.hust.edu.cn/hTFtarget#!/download), we obtained the list of all 1,342,129 TF-target regulations (file: “TF-Target-information.txt”), involving 495 TFs and 38,183 targets. These regulations are directed from TFs to their targets and do not have weights representing their confidence.

Next, we filtered the list of regulations to include only the 289 genes in our dataset, resulting in a list of 399 TF-target regulations involving 8 TFs and 191 targets. We then constructed a prior network, considering the regulations as directed edges from the TFs to their corresponding targets.

Harmonizome 3.0 database:

The Harmonizome 3.0 is a collector of genomic databases aggregating and standardizing functional genomics data and genes interactions.

From the Harmonizome 3.0 site, we retrieved TF-target regulations from three distinct sources: TRANSFAC, CHEA, and ENCODE. TRANSFAC provides 100,562 interactions, CHEA offers 386,777 interactions, and ENCODE supplies 1,655,385 interactions. Upon filtering for the 289 genes in our dataset, no interactions from TRANSFAC remained. Only 94 interactions persisted from CHEA, all involving the same 3 TFs. Consequently, both sources were disregarded as potential priors. Similarly, after filtering the same 289 genes, ENCODE yielded 914 TF-target

interactions, involving 270 genes (6 TFs and 264 targets), meeting our requirements sufficiently, and thus retained as prior knowledge.

Prior GRN	genes	edges
GIANT	232	6074
GIANT weights > 0.2	119	832
TissueNexus	271	3480
TissueNexus weights > 0.6	242	1306
STRING	181	498
ENCODE	270	914
hTFtarget	199	399

Table 2: The dimensions of the prior networks used in the inference process.

Once we had the dataset and priors from the different databases, I tested the algorithm by combining the preprocessed dataset of 289 genes and 463 cis-eQTLs with a different prior for each run. Additionally, to establish a comparison, we conducted a run without incorporating any prior network (equivalent to the FSSEM method).

The runs produced the following gene regulatory networks.

	No prior (FSSEM)		TissueNexus (>0.6)		STRING		hTFtarget	
GRN	edges	nodes	edges	nodes	edges	nodes	edges	nodes
Normal	421	169	648	248	431	203	480	198
Tumor	504	172	660	246	478	206	544	201
Differential	179	104	92	95	160	132	174	111

	ENCODE		GIANT (> 0.2)	
GRN	edges	nodes	edges	nodes
Normal	521	217	499	188
Tumor	597	223	500	191
Differential	193	115	130	102

Table 3: The table displays the dimensions of the gene regulatory networks inferred through FLAP using different priors. Three types of GRNs are considered: normal tissue (NormalGRN), tumor tissue (TumorGRN), and the difference network (DifferenceGRN) obtained from the difference between the tumor tissue network and the normal tissue network.

However, it is noteworthy that when using the GIANT and TissueNexus prior networks without filtering for edge weights, the method did not converge to a conclusive result within the defined parameters. Conversely, the subsetted versions of the GIANT and TissueNexus networks, which were filtered by gene weights above a threshold, produced results. This suggests that the effectiveness of our method may vary depending on the characteristics of the prior network used.

We observed that the prior networks utilized for our dataset exhibited minimal overlap in terms of gene interactions.

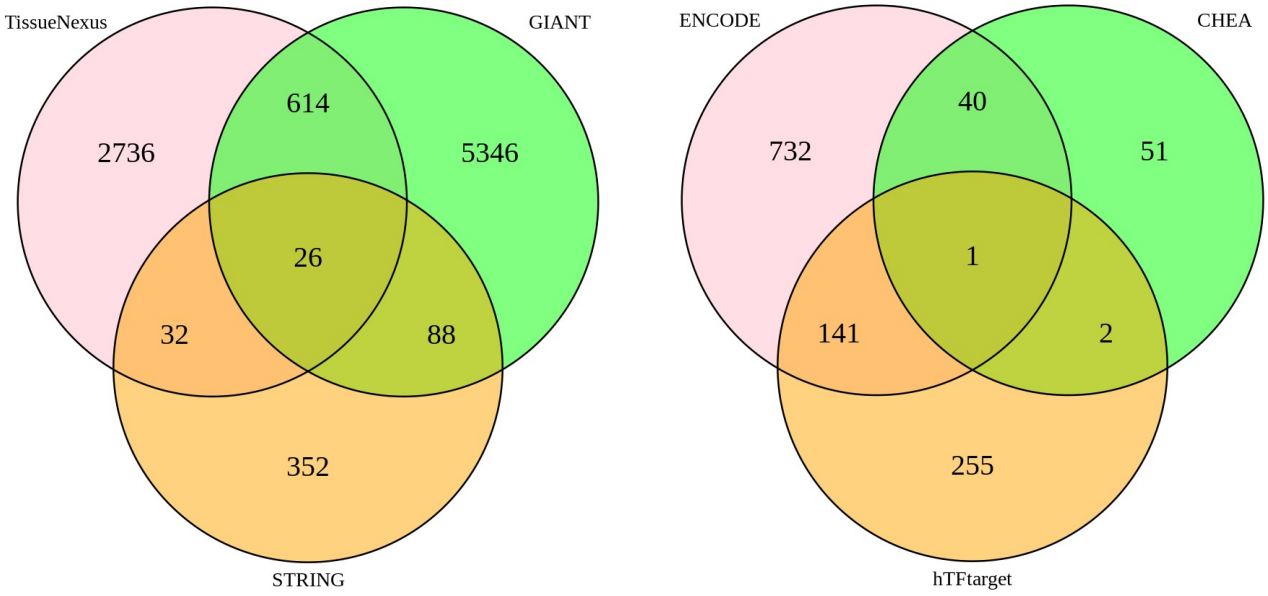

Fig. 9: This figure illustrates two Venn diagrams representing the intersection of prior gene regulatory networks obtained from various sources. The left Venn diagram shows the common edges among the prior networks from GIANT, TissueNexus, and STRING. On the right, the Venn diagram displays the common TF-target edges among ENCODE, hTFtarget, and CHEA.

Therefore, we opted to expand our prior networks by identifying gene interactions that were consistent across multiple databases: GIANT and TissueNexus (considering all edges as well as edges weighted > 0.2 and > 0.6 respectively), along with ENCODE and hTFtarget. By focusing on the intersections of these databases, we aimed to enhance the reliability of our prior information, selecting only those interactions that agreed across multiple sources.

Prior GRN (intersection)	genes	edges
GIANT, TissueNexus	154	640
GIANT (weights>0.2), TissueNexus (weights>0.6)	54	120
ENCODE, hTFtarget	152	226

Table 4: The dimensions of the prior gene regulatory networks obtained from the intersection of different databases.

Resulting in the following networks:

	No prior (FSSEM)		ENCODE ∩ hTFtarget		GIANT ∩ TissueNexus		GIANT > 0.2 ∩ TissueNexus > 0.6	
GRN	edges	nodes	edges	nodes	edges	nodes	edges	nodes
Normal	421	169	471	191	400	186	516	195
Tumor	504	172	562	189	433	192	608	193
Differential	179	104	214	121	146	117	222	119

Table 5: This table presents the dimensions of the gene regulatory networks (GRNs) inferred through FLAP using the intersection of prior networks from two different databases. Three types of GRNs are considered: normal tissue (*Normal*), tumor tissue (*Tumor*), and the difference network (*Differential*) obtained from the difference between the tumor tissue network and the normal tissue network.

3.2.1 Validation with Over-representation analysis

To validate the inferred networks and their relevance in the context of lung cancer, we performed an over-representation analysis (ORA). ORA is a type of functional analysis used to identify pathways or biological processes that are significantly enriched in a list of genes that have been identified as relevant based on prior analysis or experimental results.

This method evaluates whether predefined sets of genes representing pathways, biological processes, or functional categories are over-represented in a given list of genes more than would be expected by chance. In other words, the given list of genes is enriched for the gene sets and the biological processes they represent.

In our case, we are validating our inferred gene regulatory networks (GRNs) by evaluating if the genes present in the differential network are enriched for gene sets related to lung cancer. We used

curated gene sets from the Molecular Signature Database (MSigDB) C2 collection, which includes gene sets obtained from various sources such as online pathway databases, biomedical literature, and individual domain experts. This collection comprises 7,233 gene sets. We filtered this collection for gene sets containing the keywords “lung cancer,” “lung tumor,” “lung carcinoma,” and “LUCA” in their descriptions, resulting in a subset of 200 gene sets related to lung cancer.

Then to evaluate the over-representation, for each gene set we construct a 2x2 contigency table:

	In Gene Set	**Not in Gene Set**
In Gene list	a	b
Not in Gene List	c	d

where

a : the number of genes in the list of genes of the differential network that are also present in the gene set

b : the number of genes in the list of genes of the differential network that are not in the gene set

c : the number of genes not in the list of genes of the differential network but that are present in the gene set

d : the number of genes that are neither in the list of genes of the differential network nor in the gene set

We then applied Fisher’s exact test to determine whether the proportion of genes in the gene list that are also in the gene set is significantly different from what would be expected by chance. The test calculates the probability of observing the given overlap under the null hypothesis that the genes in the differential network gene list are randomly distributed with respect to the gene sets.

The formula for Fisher's exact test is:

$$\text{p-value} = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!}$$

where $n = a+b+c+d$ is the total number of genes considered.

The p-value obtained indicates the likelihood that the observed overlap between the gene list and the gene set is due to random chance. We chose a p-value threshold of < 0.05 to consider gene sets as significantly enriched in the gene list of our differential network (Table 6 and Table 7).

	FSSEM	FLAP				
	no prior	ENCODE	hTFtarget	STRING	TissueNexus	GIANT
Gene Set	pval	pval	pval	pval	pval	pval
SMID_BREAST_CANCER_LUMINAL_B_DN	0.00514	0.00135	0.01537	0.006444		0.03694
SMID_BREAST_CANCER_BASAL_DN		0.01629	0.01537			0.00445
HOLLERN_EMT_BREAST_TUMOR_UP	0.01615	0.02428				0.01492
CHIANG_LIVER_CANCER_SUBCLASS_POLYSOMY7_UP	0.04574			0.01906	0.04163	0.00513
ZHU_CMV_ALL_DN	0.04574					
IWANAGA_CARCINOGENESIS_BY_KRAS_PTEN_UP					0.03476	
SPIELMAN_LYMPHOBLAST_EUROPEAN_VS_ASIAN_DN						0.00072

Table 6: Significantly enriched gene sets for FSSEM (not using priors) and FLAP using prior information from a single source, namely ENCODE, hTFtarget, STRING, TissueNexus and GIANT. For each gene set we highlighted in *yellow* the best p-value.

	FSSEM	FLAP		
	no prior	GIANT ∩ TissueNexus	(GIANT > 0.2) ∩ (TissueNexus > 0.6)	ENCODE ∩ hTFtarget
Gene Set	pval	pval	pval	pval
SMID_BREAST_CANCER_LUMINAL_B_DN	0.00514	0.017057	0.001829	0.00211
SMID_BREAST_CANCER_BASAL_DN		0.017057		0.01923
HOLLERN_EMT_BREAST_TUMOR_UP	0.01615		0.027895	0.029844
CHIANG_LIVER_CANCER_SUBCLASS_POLYSOMY7_UP	0.04574		0.011256	
ZHU_CMV_ALL_DN	0.04574			
WHITEFORD_PEDIATRIC_CANCER_MARKERS		0.026044	0.027895	
KOBAYASHI_EGFR_SIGNALING_24HR_DN			0.011256	

Table 7: Significantly enriched gene sets for FSSEM (not using priors) and FLAP using prior information from an intersection of sources, GIANT and TissueNexus, GIANT with edge weights > 0.2 and TissueNexus edge weights > 0.6), ENCODE and hTFtarget. For each gene set we highlighted in *yellow* the best p-value.

Among all methods, those with more enriched gene sets are FLAP with GIANT (>0.2) and FLAP with the combined prior of GIANT (>0.2) ∩ TissueNexus (>0.6), with 5 enriched gene sets, compared to the 4 of the method without the use of priors.

This could be explained by GIANT being tissue-specific for lung and of better quality overall compared to TissueNexus, which is also specific for lung.

FLAP with prior networks from STRING, hTFtarget, and TissueNexus only had 2 enriched gene sets, suggesting they may be of lesser quality as sources of prior information.

ENCODE seems to perform only slightly worse than the method without prior, with 3 enriched gene sets. This may indicate better data quality compared to other TF-target databases like hTFtarget.

The ORA results show that the biological plausibility of the inferred networks on real lung cancer data depend on the quality of the prior knowledge used.

Compared with the original method FSSEM and its enrichment in 4 gene sets, ENCODE performed sligly worse with 3 enriched gene sets of which 1 with better p-value than FSSEM. Meanwhile GIANT performed better than FSSEM, with 5 enriched gene sets of which 3 out of 5 had also better p-value (Table 6).

The use of priors obtained through the intersection of the single ones did not achieve better results than their single prior counterparts, with GIANT ∩ Tissuenexus being the best one with 5 enriched gene sets, 2 of which had better p-value than FSSEM.

Considering the low consensus on the gene interaction among the prior knowledge sources shown in Fig. 9, these results suggest that those databases contains a high number of wrong informations about gene interactions. This underscores the importance of using high-quality prior knowledge in network inference.

3.2.2 Validation with literature

One limitation of using functional enrichment analysis, such as ORA, for the validation of inferred gene regulatory networks is that it focuses solely on the list of genes, without considering the interactions or edges between them. As a result, important regulatory relationships between genes may be overlooked, hindering the comprehensive understanding of gene regulatory networks.

To address this limitation, it is beneficial to validate the inferred gene regulatory networks by comparing them with existing literature. In our case, validation with literature can focus on the differential gene network, where relevant genes are identified as dysregulated due to the different sets of edges that connect them in the tumor and control cases.

To validate our results, we focused on the genes within the inferred differential networks, as these are the ones identified as dysregulated. We ranked these genes based on their degree, which represents the number of edges connected to each gene and is considered a measure of the level of

their dysregulation. Genes with more differential edges are considered more dysregulated. Subsequently, we extracted the top ten ranked genes as a signature of each network. We then conducted a literature search using PubMed to evaluate the involvement of these signature genes in "Lung Adenocarcinoma" (LUAD) or "Non-Small Cell Lung Cancer" (NSCLC), given that the dataset focuses on patients affected by LUAD, which is a type of NSCLC.

Here in Table 8 we summarize the research in literature for the differential networks. In the first column are listed the resulting network obtained with each prior, in the second column we report how many of the genes in the top ten signatures have been reported to be involved with LUAD or NSCLC. In the third column is the total number of relevant papers involving those genes.

Method	LUAD/NSCLC Publication Count for Top 10 Genes	Total Publications
no prior (FSSEM)	7	9
GIANT > 0.2	8	22
TissueNexus > 0.6	8	23
STRING	7	24
ENCODE	7	12
hTFtarget	8	12
TissueNexus ∩ GIANT	8	24
TissueNexus >0.6 ∩ GIANT > 0.2	7	20
ENCODE ∩ hTFtarget	7	9

Table 8: Summary of the validation with literature for the top ten signature genes.

In the following tables (Table 9 - Table 16), we present in detail the top ten signature genes for each differential network. The list of relevant publications can be found in the supplementary material S1.

FSSEM	Prior : no prior		
entrezID	gene	degree	LUAD/NSCLC Publications count
3120	HLA-DQB2	19	1
84072	TPSAB1	13	0
7177	HORMAD1	13	2
79782	CLIC2	12	0
1193	LRRC31	12	1
5450	POU2AF1	11	1
5950	RBP4	11	1
105375355	MEOX2	10	2
4223	UPK3B	10	0
9957	HS3ST1	9	1

Table 9: Top 10 signature genes for the FSSEM method, equivalent to FLAP without prior network. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC.

FLAP	Prior: GIANT (>0.2)		
entrezID	gene	degree	LUAD/NSCLC Publications count
705	BYSL	9	0
3120	HLA-DQB2	9	1
847	CAT	8	5
55120	FANCL	8	1
23433	RHOQ	8	1
6790	AURKA	7	9
79782	LRRC31	7	1
10051	SMC4	7	3
11034	DSTN	6	1
8853	ASAP2	6	0

Table 10: Top 10 signature genes for the FLAP method with a prior network from GIANT with edge weights above 0.2. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior: TissueNexus (>0.6)		
entrezID	gene	degree	LUAD/NSCLC Publications count
7048	TGFBR2	8	1
5450	POU2AF1	6	1
2358	FPR2	6	3
3120	HLA-DQB2	5	1
79782	LRRC31	5	1
84072	HORMAD1	4	2
1193	CLIC2	4	0
105375355	UPK3B	4	0
6790	AURKA	4	9
847	CAT	4	5

Table 11: Top 10 signature genes for the FLAP method with a prior network from TissueNexus with edge weights above 0.6. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior : STRING		
entrezID	gene	degree	LUAD/NSCLC Publications count
79782	LRRC31	14	1
3120	HLA-DQB2	13	1
5450	POU2AF1	9	1
84072	HORMAD1	8	2
57480	PLEKHG1	7	2
54915	YTHDF1	7	14
2358	CLIC2	6	0
1193	TPSAB1	6	0
105375355	FPR2	6	3
140686	WFDC3	6	0

Table 12: Top 10 signature genes for the FLAP method with a prior network from STRING. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior: ENCODE		
entrezID	gene	degree	LUAD/NSCLC Publications count
1997	ELF1	27	4
5929	RBBP5	23	0
9682	KDM4A	22	1
140686	WFDC3	13	0
467	ATF3	12	2
3120	HLA-DQB2	12	1
79782	LRRC31	12	1
1193	CLIC2	12	0
5450	POU2AF1	10	1
84072	HORMAD1	10	2

Table 13: Top 10 signature genes for the FLAP method with a prior network from ENCODE. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior : hTFtarget		
entrezID	gene	degree	LUAD/NSCLC Publications count
467	ATF3	18	2
3120	HLA-DQB2	15	1
57332	CBX8	14	1
79782	LRRC31	11	1
1193	CLIC2	11	0
5450	POU2AF1	9	1
84072	HORMAD1	9	2
5929	RBBP5	8	0
57214	CEMIP	8	1
2358	FPR2	7	3

Table 14: Top 10 signature genes for the FLAP method with a prior network from hTFtarget. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior: GIANT ∩ TissueNexus		
entrezID	**gene**	**degree**	**LUAD/NSCLC Publications count**
6790	AURKA	10	9
79782	LRRC31	9	1
10051	SMC4	9	3
1193	CLIC2	8	0
3120	HLA-DQB2	8	1
5450	POU2AF1	7	1
84072	HORMAD1	7	2
8607	RUVBL1	7	6
705	BYSL	7	0
9631	NUP155	6	1

Table 15: Top 10 signature genes for the FLAP method with a prior network from the intersection of the prior networks of GIANT and TissueNexus. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior: GIANT (>0.2) ∩ TissueNexus (>0.6)		
entrezID	**gene**	**degree**	**LUAD/NSCLC Publications count**
3120	HLA-DQB2	18	1
84072	HORMAD1	14	2
7177	TPSAB1	13	0
1193	CLIC2	13	0
79782	LRRC31	12	1
57214	CEMIP	12	1
5450	POU2AF1	12	1
6790	AURKA	11	9
105375355	UPK3B	10	0
847	CAT	10	5

Table 16: Top 10 signature genes for the FLAP method with a prior network from the intersection of the prior networks of GIANT (edge weights above 0.2) and TissueNexus (edge weights above 0.6). For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

FLAP	Prior: hTFtarget ∩ ENCODE		
entrezID	gene	degree	LUAD/NSCLC Publications count
3120	HLA-DQB2	19	1
467	ATF3	16	2
84072	HORMAD1	14	2
7177	TPSAB1	13	0
1193	CLIC2	12	0
79782	LRRC31	12	1
5950	RBP4	12	1
5450	POU2AF1	11	1
57214	CEMIP	10	1
105375355	UPK3B	10	0

Table 17: Top 10 signature genes for the FLAP method with a prior network from the intersection of the prior networks of hTFtarget and ENCODE. For each gene, we list their Entrez ID, gene name, degree in the differential network, and the number of publications involving the gene with LUAD or NSCLC. Genes that also appear in the top 10 signature genes of FSSEM are highlighted in *yellow*.

We observed that our FLAP method identified several signature genes relevant to LUAD or NSCLC, comparable to those identified by the FSSEM method. Notably, FLAP revealed some new genes not found by FSSEM, indicating that different sources of prior knowledge can uncover new insights into gene regulations from the same data. For instance, the genes AURKA, identified using functional gene networks like GIANT and TissueNexus, has 9 publications linking it to lung cancer. Similarly, YTHDF1 was associated with lung cancer in 14 studies when the prior network from STRING was used. Additionally, the transcription factors ATF and ELF1 were identified when using prior networks from transcription factor databases. These findings suggest that the choice of prior knowledge significantly influences the identification of relevant genes.

Chapter 4: Conclusions and Discussion

This study addresses the challenges inherent in inferring gene regulatory networks (GRNs) and proposes a novel approach, Fused Lasso Adaptive Prior (FLAP), to enhance the inference process by integrating prior knowledge of gene interactions.

In Chapter 1, we reviewed GRN inference methods developed over the last two decades, aiming to understand the challenges of the problem and the efforts made to overcome limitations. We discussed how the DREAM5 challenge marked a pivotal point after which modern state-of-the-art inference methods emerged. Among these, Structural Equation Models (SEM) became a successful framework, allowing the use of multi-omics, perturbations, and relatively straightforward regularized linear models with machine learning techniques.

In Chapter 2: Methods, we explained how SEM could be used to infer GRNs and how linear regression methods were used to incorporate assumptions to better reflect the biological nature of the problem. We then introduced the Fused Lasso Adaptive Prior (FLAP) method, an extension of Fused Sparse SEM (FSSEM), and demonstrated how it incorporates prior knowledge of gene interactions to guide the GRN inference process.

In Chapter 3, we tested FLAP on synthetic and real data. We used synthetic data to address two challenges regarding the integration of imperfect prior knowledge. The first challenge involved the calibration of penalty factors to encode the prior knowledge. We concluded that the most effective way to integrate prior knowledge was to set the penalty factors to 0, thereby always including the known gene interactions in the initial estimate. This ensures that the initial estimate can guide the second step of feature selection based on data and prior knowledge.

In the second challenge, we compared the effectiveness of integrating prior knowledge in the ridge regression step with its integration in the adaptive fused lasso step. Our results indicated that integrating prior knowledge in the ridge regression step is more robust, especially by having better precision when the prior knowledge presents wrong gene interactions.

We then tested FLAP robustness by comparing its performance with noise in the data and noise in the prior network. To assess this, we used data with an increased level of noise along with a perfect prior network and a randomly rewired prior network with the same connectivity. Our results showed that while noise in the data degrades the performance of FLAP, its performance degraded twice as fast with the randomly rewired prior network compared to the perfect prior network. This indicates

that FLAP performance is not solely dependent on the characteristic connectivity of the prior network but rather on the information contained in the prior network.

Additionally, we compared our method, FLAP, with two other existing methods, FSSEM and BFDSEM, under different sample size conditions. Our results showed that FLAP has the potential to outperform FSSEM in terms of precision for various sample sizes, while maintaining similar recall values. Although BFDSEM showed promising performance with increased sample sizes, our method consistently outperformed both BFDSEM and FSSEM in terms of precision.

To evaluate the biological relevance of the inferred networks, we utilized a dataset from the GEO database and integrated multiple sources of prior knowledge to guide the gene network inference process. Our results demonstrated that the quality of the inferred networks depends significantly on the quality of the prior knowledge used. Using high-quality prior networks, such as GIANT and ENCODE, resulted in more enriched gene sets related to lung cancer in the over-representation analysis (ORA).

Furthermore, we validated FLAP's inferred networks by conducting a literature search on the top 10 signature genes. Our results showed that a similar number of signature genes were relevant for lung cancer in FLAP as in FSSEM. However, FLAP identified additional signature genes relevant to lung cancer that were not present in the signature genes identified by FSSEM. This suggests that different prior knowledge sources may reveal new information about the same data.

While our FLAP method shows promising results in inferring gene regulatory networks, it is essential to acknowledge its limitations.

One of the main challenges inherent in utilizing prior knowledge for network inference is the quality of the available information. While our method showed some tolerance for incorrect information when tested on synthetic data (Chapter 3.1), the analysis on real data (Chapter 3.2) revealed that databases often have little agreement on known gene interactions. This lack of consensus results in some prior knowledge sources containing a significant amount of erroneous information, surpassing the tolerance level of our FLAP method. Consequently, this leads to worse results compared to the data-driven FSSEM method.

Another limitation is that FLAP does not utilize the weights of the edges in the prior networks. The penalization of the edges in the networks is uniform, and all edges in the prior are not penalized and are included in the initial estimate. This may not be optimal because it considers all interactions to be equally important.

Incorporating the weights provided by different databases is challenging due to the variety of metrics used. Some databases use measures of strength, others use confidence levels of the interactions, while some simply provide a ranking of interactions. harmonizing these different metrics into a unified framework remains a complex task.

Future research may address these issues by focusing on enhancing the quality of the prior knowledge. One approach could involve creating a consensus among multiple high-confidence databases. Establishing a consensus level would allow for the refinement of penalization by implementing a non-uniform penalization of the prior edges, with edges being more or less penalized based on their confidence level.

Another future extension of our work could involve testing FLAP on different types of data that combine gene expression and perturbations. For example, CROP-seq datasets combine single-cell RNA sequencing (scRNAseq) with CRISPR interference (CRISPRi) or RNA-seq with eQTLs.

Furthermore, exploring FLAP's performance on datasets related to various diseases and drug sensitivity would provide further insights into its applicability and robustness across different experimental setups.

BIBLIOGRAPHY

[1] G. Stolovitzky, R. J. Prill, and A. Califano, “Lessons from the DREAM2 challenges: A community effort to assess biological network inference,” *Ann. N. Y. Acad. Sci.*, vol. 1158, pp. 159–195, 2009, doi: 10.1111/j.1749-6632.2009.04497.x.

[2] A. Madar, A. Greenfield, E. Vanden-Eijnden, and R. Bonneau, “DREAM3: Network inference using dynamic context likelihood of relatedness and the inferelator,” *PLoS One*, vol. 5, no. 3, 2010, doi: 10.1371/journal.pone.0009803.

[3] A. Greenfield, A. Madar, H. Ostrer, and R. Bonneau, “DREAM4: Combining genetic and dynamic information to identify biological networks and Dynamical Models,” *PLoS One*, vol. 5, no. 10, 2010, doi: 10.1371/journal.pone.0013397.

[4] D. Marbach *et al.*, “Wisdom of crowds for robust gene network inference the DREAM5 Consortium HHS Public Access,” *Nat Methods*, vol. 9, no. 8, pp. 796–804, 2016, doi: 10.1038/nmeth.2016.Wisdom.

[5] X. Zhou and X. Cai, “Inference of differential gene regulatory networks based on gene expression and genetic perturbation data,” *Bioinformatics*, vol. 36, no. 1, pp. 197–204, 2020, doi: 10.1093/bioinformatics/btz529.

[6] A. A. Margolin *et al.*, “ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context,” *BMC Bioinformatics*, vol. 7, no. SUPPL.1, pp. 1–15, 2006, doi: 10.1186/1471-2105-7-S1-S7.

[7] J. J. Faith *et al.*, “Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles,” *PLoS Biol.*, vol. 5, no. 1, pp. 0054–0066, 2007, doi: 10.1371/journal.pbio.0050008.

[8] P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi, “Information-theoretic inference of large transcriptional regulatory networks,” *Eurasip J. Bioinforma. Syst. Biol.*, vol. 2007, no. i, 2007, doi: 10.1155/2007/79879.

[9] G. Altay and F. Emmert-Streib, “Revealing differences in gene network inference algorithms on the network level by ensemble methods,” *Bioinformatics*, vol. 26, no. 14, pp. 1738–1744, 2010, doi: 10.1093/bioinformatics/btq259.

[10] N. Friedman, M. Linial, I. Nachman, and D. Pe’er, “Using Bayesian networks to analyze expression data,” *J. Comput. Biol.*, vol. 7, no. 3–4, pp. 601–620, 2000, doi: 10.1089/106652700750050961.

[11] F. Markowetz and R. Spang, “Inferring cellular networks - A review,” *BMC Bioinformatics*, vol. 8, no. SUPPL. 6, 2007, doi: 10.1186/1471-2105-8-S6-S5.

[12] Z. Liu, B. Malone, and C. Yuan, “Empirical evaluation of scoring functions for Bayesian network model selection.,” *BMC Bioinformatics*, vol. 13 Suppl 15, no. Suppl 15, p. S14, 2012, doi: 10.1186/1471-2105-13-S15-S14.

[13] D. Husmeier, “Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks,” *Bioinformatics*, vol. 19, no. 17, pp. 2271–2282, 2003, doi: 10.1093/bioinformatics/btg313.

[14] T. S. Gardner, D. Di Bernardo, D. Lorenz, and J. J. Collins, “Inferring genetic networks and identifying compound mode of action via expression profiling,” *Science (80-.).*, vol. 301, no. 5629, pp. 102–105, 2003, doi: 10.1126/science.1081900.

[15] J. Tegnér, M. K. S. Yeung, J. Hasty, and J. J. Collins, “Reverse engineering gene networks: Integrating genetic perturbations with dynamical modeling,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 100, no. 10, pp. 5944–5949, 2003, doi: 10.1073/pnas.0933416100.

[16] M. A. Efroymson, *Mathematical methods for digital computers, Ch. Multiple Regression Analysis*. New York: John Wiley.

[17] B. Efron *et al.*, “Least angle regression,” *Ann. Stat.*, vol. 32, no. 2, pp. 407–499, 2004, doi: 10.1214/009053604000000067.

[18] A. E. Hoerl and R. W. Kennard, “Ridge Regression: Biased Estimation for Nonorthogonal Problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970, doi: 10.1080/00401706.1970.10488634.

[19] R. Tibshirani, “Regression Shrinkage and Selection Via the Lasso,” *J. R. Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996, doi: 10.1111/j.2517-6161.1996.tb02080.x.

[20] V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts, “Inferring regulatory networks from expression data using tree-based methods,” *PLoS One*, vol. 5, no. 9, pp. 1–10, 2010, doi: 10.1371/journal.pone.0012776.

[21] V. Filkov, *Handbook of computational molecular biology, Ch. Identifying Gene Regulatory Networks from Gene Expression Data*. Chapman & Hall/CRC Computer and Information Science Series, 2005.

[22] N. D. Lawrence, G. Sanguinetti, and M. Rattray, “Modelling transcriptional regulation using Gaussian processes,” *NIPS 2006 Proc. 19th Int. Conf. Neural Inf. Process. Syst.*, pp. 785–792, 2006, doi: 10.7551/mitpress/7503.003.0103.

[23] P. Gao, A. Honkela, M. Rattray, and N. D. Lawrence, “Gaussian process modelling of latent chemical species: Applications to inferring transcription factor activities,” *Bioinformatics*, vol. 24, no. 16, pp. 70–75, 2008, doi: 10.1093/bioinformatics/btn278.

[24] T. Äijö and H. Lähdesmäki, “Learning gene regulatory networks from gene expression measurements using non-parametric molecular kinetics,” *Bioinformatics*, vol. 25, no. 22, pp. 2937–2944, 2009, doi: 10.1093/bioinformatics/btp511.

[25] I. Rauluseviciute *et al.*, “JASPAR 2024: 20thãnniversary of the open-access database of transcription factor binding profiles,” *Nucleic Acids Res.*, vol. 52, no. D1, pp. D174–D182, 2024, doi: 10.1093/nar/gkad1059.

[26] E. Wingender, “The TRANSFAC project as an example of framework technology that supports the analysis of genomic regulation,” *Brief. Bioinform.*, vol. 9, no. 4, pp. 326–332, 2008, doi: 10.1093/bib/bbn016.

[27] F. Abascal *et al.*, “Expanded encyclopaedias of DNA elements in the human and mouse genomes,” *Nature*, vol. 583, no. 7818, pp. 699–710, 2020, doi: 10.1038/s41586-020-2493-4.

[28] A. B. Keenan *et al.*, “ChEA3: transcription factor enrichment analysis by orthogonal omics integration,” *Nucleic Acids Res.*, vol. 47, no. W1, pp. W212–W224, 2019, doi: 10.1093/nar/gkz446.

[29] C. Skok Gibbs *et al.*, “High-performance single-cell gene regulatory network inference at scale: The Inferelator 3.0,” *Bioinformatics*, vol. 38, no. 9, pp. 2519–2528, 2022, doi: 10.1093/bioinformatics/btac117.

[30] K. Kamimoto, B. Stringa, C. M. Hoffmann, K. Jindal, L. Solnica-Krezel, and S. A. Morris, “Dissecting cell identity via network inference and in silico gene perturbation,” *Nature*, vol. 614, no. 7949, pp. 742–751, 2023, doi: 10.1038/s41586-022-05688-9.

[31] S. Aibar *et al.*, “Europe PMC Funders Group Europe PMC Funders Author Manuscripts SCENIC : Single-cell regulatory network inference and clustering,” *Nat. Methods*, vol. 14, no. 11, pp. 1083–1086, 2018, doi: 10.1038/nmeth.4463.SCENIC.

[32] S. Roy, S. Lagree, Z. Hou, J. A. Thomson, R. Stewart, and A. P. Gasch, “Integrated Module and Gene-Specific Regulatory Inference Implicates Upstream Signaling Networks,” *PLoS Comput. Biol.*, vol. 9, no. 10, 2013, doi: 10.1371/journal.pcbi.1003252.

[33] K. Glass, C. Huttenhower, J. Quackenbush, and G. C. Yuan, “Passing Messages between Biological Networks to Refine Predicted Interactions,” *PLoS One*, vol. 8, no. 5, 2013, doi: 10.1371/journal.pone.0064832.

[34] D. Szklarczyk *et al.*, “STRING v11: Protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets,” *Nucleic Acids Res.*, vol. 47, no. D1, pp. D607–D613, 2019, doi: 10.1093/nar/gky1131.

[35] D. Seçilmiş, T. Hillerton, A. Tjärnberg, S. Nelander, T. E. M. Nordling, and E. L. L. Sonnhammer, “Knowledge of the perturbation design is essential for accurate gene regulatory network inference,” *Sci. Rep.*, vol. 12, no. 1, pp. 1–12, 2022, doi: 10.1038/s41598-022-19005-x.

[36] J. Zhu *et al.*, “Increasing the power to detect causal associations by combining genotypic and expression data in segregating populations,” *PLoS Comput. Biol.*, vol. 3, no. 4, pp. 692–703, 2007, doi: 10.1371/journal.pcbi.0030069.

[37] E. C. Neto, C. T. Ferrara, A. D. Attie, and B. S. Yandell, “Inferring causal phenotype networks from segregating populations,” *Genetics*, vol. 179, no. 2, pp. 1089–1100, 2008, doi: 10.1534/genetics.107.085167.

[38] M. Xiong, J. Li, and X. Fang, “Identification of Genetic Networks,” *Genetics*, vol. 166, no. 2, pp. 1037–1052, 2004, doi: 10.1534/genetics.166.2.1037.

[39] B. Liu, A. De La Fuente, and I. Hoeschele, “Gene network inference via structural equation modeling in genetical genomics experiments,” *Genetics*, vol. 178, no. 3, pp. 1763–1776, 2008, doi: 10.1534/genetics.107.080069.

[40] B. A. Logsdon and J. Mezey, “Gene expression network reconstruction by convex feature selection when incorporating genetic perturbations,” *PLoS Comput. Biol.*, vol. 6, no. 12, 2010, doi: 10.1371/journal.pcbi.1001014.

[41] X. Cai, J. A. Bazerque, and G. B. Giannakis, “Inference of Gene Regulatory Networks with Sparse Structural Equation Models Exploiting Genetic Perturbations,” *PLoS Comput. Biol.*, vol. 9, no. 5, 2013, doi: 10.1371/journal.pcbi.1003068.

[42] Y. Li, D. Liu, T. Li, and Y. Zhu, “Bayesian differential analysis of gene regulatory networks exploiting genetic perturbations,” *BMC Bioinformatics*, vol. 21, no. 1, pp. 1–13, 2020, doi: 10.1186/s12859-019-3314-3.

[43] R. Oughtred *et al.*, “The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions,” *Protein Sci.*, vol. 30, no. 1, pp. 187–200, 2021, doi: 10.1002/pro.3978.

[44] Y. Yi, Y. Fang, K. Wu, Y. Liu, and W. Zhang, “Comprehensive gene and pathway analysis of cervical cancer progression,” *Oncol. Lett.*, vol. 19, no. 4, pp. 3316–3332, 2020, doi: 10.3892/ol.2020.11439.

[45] M. Milacic *et al.*, “The Reactome Pathway Knowledgebase 2024,” *Nucleic Acids Res.*, vol. 52, no. D1, pp. D672–D678, 2024, doi: 10.1093/nar/gkad1025.

[46] S. A. Aleksander *et al.*, “The Gene Ontology knowledgebase in 2023,” *Genetics*, vol. 224, no. 1, pp. 1–14, 2023, doi: 10.1093/genetics/iyad031.

[47] A. Chatr-aryamontri *et al.*, “MINT: The Molecular INTeraction database,” *Nucleic Acids Res.*, vol. 35, no. SUPPL. 1, pp. 2006–2008, 2007, doi: 10.1093/nar/gkl950.

[48] S. Peri *et al.*, “Development of human protein reference database as an initial platform for approaching systems biology in humans,” *Genome Res.*, vol. 13, no. 10, pp. 2363–2371, 2003, doi: 10.1101/gr.1680803.

[49] J. Amberger, C. Bocchini, and A. Hamosh, “A new face and new challenges for Online Mendelian Inheritance in Man (OMIM®),” *Hum. Mutat.*, vol. 32, no. 5, pp. 564–567, 2011, doi: 10.1002/humu.21466.

[50] A. D. Rouillard *et al.*, “The harmonizome: a collection of processed datasets gathered to serve and mine knowledge about genes and proteins,” *Database (Oxford).*, vol. 2016, pp. 1–16, 2016, doi: 10.1093/database/baw100.

[51] K. Daily, V. R. Patel, P. Rigor, X. Xie, and P. Baldi, “MotifMap: Integrative genome-wide maps of regulatory motif sites for model species,” *BMC Bioinformatics*, vol. 12, no. 1, 2011, doi: 10.1186/1471-2105-12-495.

[52] A. Liberzon *et al.*, “The Molecular Signatures Database (MSigDB) hallmark gene set collection,” vol. 1, no. 6, pp. 417–425, 2016, doi: 10.1016/j.cels.2015.12.004.The.

[53] C. S. Greene *et al.*, “Understanding multicellular function and disease with human tissue-specific networks,” *Nat. Genet.*, vol. 47, no. 6, pp. 569–576, 2015, doi: 10.1038/ng.3259.

[54] C. X. Lin, H. D. Li, C. Deng, Y. Guan, and J. Wang, “TissueNexus: A database of human tissue functional gene networks built with a large compendium of curated RNA-seq data,” *Nucleic Acids Res.*, vol. 50, no. D1, pp. D710–D718, 2022, doi: 10.1093/nar/gkab1133.

[55] The GTEx Consortium, “The Genotype-Tissue Expression (GTEx) project The GTEx Consortium* Abstract,” *Database Natl. Cent. Biomed. Inf.*, vol. 45, no. 6, pp. 580–585, 2013, doi: 10.1038/ng.2653.The.

[56] A. A. Shabalin, “Matrix eQTL: Ultra fast eQTL analysis via large matrix operations,” *Bioinformatics*, vol. 28, no. 10, pp. 1353–1358, 2012, doi: 10.1093/bioinformatics/bts163.

[57] J. H. Degnan, E. and Schelar, and J. Liu, “Genomics and genome-wide association studies: An integrative approach for expression QTL mapping,” *Bone*, vol. 23, no. 1, pp. 1–7, 2008, doi: 10.1016/j.ygeno.2008.05.012.Genomics.

[58] J. Listgarten, C. Kadie, E. E. Schadt, and D. Heckerman, “Correction for hidden confounders in the genetic analysis of gene expression,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 107, no. 38, pp. 16465–16470, 2010, doi: 10.1073/pnas.1002425107.

[59] H. Zou, “The adaptive lasso and its oracle properties,” *J. Am. Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006, doi: 10.1198/016214506000000735.

[60] V. Viallon, S. Lambert-Lacroix, H. Hoefling, and F. Picard, “On the robustness of the generalized fused lasso to prior specifications,” *Stat. Comput.*, vol. 26, no. 1–2, pp. 285–301, 2016, doi: 10.1007/s11222-014-9497-6.

[61] H. Höfling, H. Binder, and M. Schumacher, “A coordinate-wise optimization algorithm for the Fused Lasso,” no. Zou 2006, p. 25, 2010, [Online]. Available: http://arxiv.org/abs/1011.6409

[62] T. Pock, “Inertial Proximal Alternating Linearized Minimization (iPALM) for Nonconvex and Nonsmooth Problems arXiv : 1702 . 02505v1 [math . OC] 8 Feb 2017,” no. 640156, pp. 1–36.

[63] G. Jurman, R. Visintainer, and C. Furlanello, “An introduction to spectral distances in networks,” *Front. Artif. Intell. Appl.*, vol. 226, pp. 227–234, 2011, doi: 10.3233/978-1-60750-692-8-227.

[64] G. YU, Y. CHEN, and Y. GUO, “Design of integrated system for heterogeneous network query terminal,” *J. Comput. Appl.*, vol. 29, no. 8, pp. 2191–2193, 2009, doi: 10.3724/sp.j.1087.2009.02191.

[65] T. P. Lu *et al.*, “Integrated analyses of copy number variations and gene expression in lung adenocarcinoma,” *PLoS One*, vol. 6, no. 9, pp. 1–11, 2011, doi: 10.1371/journal.pone.0024829.

[66] L. Gautier, L. Cope, B. M. Bolstad, and R. A. Irizarry, “Affy - Analysis of Affymetrix GeneChip data at the probe level,” *Bioinformatics*, vol. 20, no. 3, pp. 307–315, 2004, doi: 10.1093/bioinformatics/btg405.

[67] M. Dai *et al.*, “Evolving gene/transcript definitions significantly alter the interpretation of GeneChip data,” *Nucleic Acids Res.*, vol. 33, no. 20, pp. 1–9, 2005, doi: 10.1093/nar/gni179.

[68] R. A. Irizarry *et al.*, “Exploration, normalization, and summaries of high density oligonucleotide array probe level data,” *Sel. Work. Terry Speed*, pp. 601–616, 2012, doi: 10.1007/978-1-4614-1347-9_15.

[69] Vaish *et al.*, “Mapping Identifiers for the Integration of Genomic Datasets with the R/Bioconductor package biomaRt,” *Bone*, vol. 23, no. 1, pp. 1–7, 2008, doi: 10.1038/nprot.2009.97.Mapping.

[70] V. Wimmer, T. Albrecht, H. J. Auinger, and C. C. Schön, “Synbreed: A framework for the analysis of genomic prediction data using R,” *Bioinformatics*, vol. 28, no. 15, pp. 2086–2087, 2012, doi: 10.1093/bioinformatics/bts335.

[71] S. R. Browning and B. L. Browning, “Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering,” *Am. J. Hum. Genet.*, vol. 81, no. 5, pp. 1084–1097, 2007, doi: 10.1086/521987.

[72] D. Altshuler *et al.*, “A haplotype map of the human genome,” *Nature*, vol. 437, no. 7063, pp. 1299–1320, 2005, doi: 10.1038/nature04226.

[73] M. E. Ritchie *et al.*, “Limma powers differential expression analyses for RNA-sequencing and microarray studies,” *Nucleic Acids Res.*, vol. 43, no. 7, p. e47, 2015, doi: 10.1093/nar/gkv007.

SUPPLEMENTARY MATERIAL

Supplementary Material S1: Top Genes Related to Lung Adenocarcinoma (LUAD) and Non-Small Cell Lung Cancer (NSCLC) with literature validation

In this supplementary material, we provide a combined list of genes identified as top regulators in the differential gene regulatory networks (GRNs) associated with Lung Adenocarcinoma (LUAD) and Non-Small Cell Lung Cancer (NSCLC), This list is part of the section 3.2.2 “Validation with Literature”, where we include references to studies and publications that link these genes to lung cancer.

[Top Genes]

HLA-DQB2:

HLA-DQB2 was identified as one of the genes that becomes upregulated during the early invasion of LUAD, indicating an enhancement of antigen presentation ability during this stage of cancer evolution [1].

HORMAD1:

HORMAD1 plays a significant role in promoting LUAD progression by inducing epithelial-mesenchymal transition (EMT) and activating the Wnt/β-catenin pathway, thereby enhancing lung cancer cell proliferation, migration, and invasion [2].

HORMAD1 is associated with resistance to oxidative stress and promotion of homologous recombination (HR) in NSCLC and LUAD. HORMAD1 expression specifies a subtype of LUAD that has adapted to mitigate DNA damage, suggesting that HORMAD1 could be a potential therapeutic target to enhance sensitivity to DNA-damaging agents or as an immunotherapeutic target in patients with NSCLC and LUAD [3].

LRRC31:

LRRC31 is identified as a downstream target gene regulated by β-hydroxybutyrate dehydrogenase 1 (BDH1) in LUAD. LRRC31 is implicated in LUAD progression through the H3K9bhb/LRRC31 axis, and targeting BDH1-high-expressing LUAD, including LRRC31, could be a potential therapeutic strategy for LUAD treatment [4].

POU2AF1:

POU2AF1 is identified as an inducible regulatory T (iTreg)-related gene associated with LUAD prognosis. POU2AF1, along with other iTreg-related genes, is part of a prognostic signature used to categorize patients into high- and low-risk subgroups. Patients with lower expression of POU2AF1 and other signature genes exhibit better prognosis and possibly greater sensitivity to traditional chemotherapy, suggesting POU2AF1 as a potential therapeutic target for LUAD treatment [5].

RBP4:

RBP4, is found to be positively associated with the risk of NSCLC. Serum RBP4 levels are significantly higher in NSCLC patients compared to healthy controls, suggesting RBP4 as a potential biomarker for NSCLC risk [6].

MEOX2:

MEOX2 is identified as a gene associated with chemoresistance and poor survival in NSCLC patients. Despite the absence of copy number variations (CNVs), MEOX2 is overexpressed and correlates with poor survival and chemoresistance in NSCLC. Its overexpression is significantly dependent on decreased levels of the repressive histone mark H3K27me3, suggesting its potential as a clinical marker for chemotherapy failure in NSCLC patients [7].

MEOX2, is identified as a key factor in cancer drug resistance and poor clinical prognosis in NSCLC patients. MEOX2 occupies the GLI-1 gene promoter region, promoting cancer drug resistance and tumor progression. Silencing MEOX2 reduces cellular resistance to cisplatinum and inhibits cellular migration and proliferation. Elevated MEOX2-dependent GLI-1 protein expression

is associated with poorer overall survival in NSCLC patients undergoing platinum-based therapy. Therefore, MEOX2 may serve as a potential therapeutic target and prognostic marker in NSCLC treatment [8].

HS3ST1:

HS3ST1 is implicated in the progression of lung adenocarcinoma (LUAD) by promoting glycolysis. HS3ST1 interacts with Glypican 4 (GPC4) to promote glycolysis, while hypoxia-derived exosomal long non-coding RNA (lncRNA) OIP5-AS1 enhances glycolysis in LUAD cells by regulating miR-200c-3p. This leads to increased LUAD cell proliferation, metastasis, and tumor size. Therefore, HS3ST1 plays a significant role in promoting LUAD progression through its involvement in glycolysis regulation mediated by hypoxia-derived exosomal lncRNA OIP5-AS1 [9].

AURKA:

AURKA overexpression is linked to poor prognosis and increased radiotherapy resistance in NSCLC. The AURKA-CXCL5 axis is identified as a crucial regulator of radiosensitivity and autophagy in NSCLC, providing potential therapeutic targets for combating NSCLC resistance to radiotherapy [10].

AURKA, along with other genes (KIAA0101, CDC20, MKI67, CHEK1, HJURP, and OIP5), is identified as a critical gene in the development and prognosis of NSCLC. The study suggests that these genes may serve as potential prognostic biomarkers and therapeutic targets for NSCLC [11].

The expression of the mitosis-associated genes AURKA, DLGAP5, TPX2, KIF11, and CKAP5 is associated with poor prognosis in NSCLC patients. AURKA, in particular, is identified as a significant prognostic marker for NSCLC [12].

AURKA involvement in non-small cell lung carcinoma (NSCLC) was highlighted in a study investigating the association between the expression profiles of mitotic spindle genes, including Aurora kinases (AURKA, AURKB, and AURKC), and clinicopathological characteristics in NSCLC patients [13]. The study revealed that increased AURKA expression is significantly

associated with poor prognosis in NSCLC patients, suggesting its potential as a therapeutic target for NSCLC treatment.

Increased AURKA expression was found to correlate with decreased time to progression and overall survival in NSCLC patients. AURKA inhibition using MLN8237 (Alisertib) reduced cell growth, especially in P53-competent NSCLC cells. Additionally, combining AURKA inhibition with radiotherapy delayed tumor growth significantly in a mouse model. These findings suggest that AURKA may be a promising therapeutic target for NSCLC, particularly when combined with radiotherapy [14].

AURKA was identified as a central player in epithelial-to-mesenchymal transition (EMT), invasion, stemness, and drug resistance in NSCLC. Using a lung tumor tissue model, the researchers found evidence suggesting a correlation between AURKA and drug resistance in cells harboring KRASG12C or EGFR mutations. In silico analysis identified AURKA as a hub linking EMT, proliferation, apoptosis, LKB1, and c-MYC. Experimental testing identified an AURKA inhibitor as a promising candidate for targeted combination therapy in KRASG12C mutant lung cancer models [15].

AURKA was identified as a potential disease gene associated with NSCLC. Using gene expression profiling and network analysis, AURKA was selected multiple times in the shortest path analysis, indicating its potential significance in NSCLC pathogenesis. These findings offer new insights into NSCLC development and may guide the development of novel therapeutic strategies [16].

AURKA was identified as one of the top 10 differentially expressed genes associated with nNSCLC. High expression of AURKA was found to be significantly correlated with poorer overall survival in NSCLC patients. Drug target analysis suggested the potential use of specific antineoplastic agents to reverse the expression of these DEGs in NSCLC patients [17].

AURKA was identified as a critical gene associated with NSCLC diagnosis and prognosis. AURKA, along with other genes such as BIRC5, CCNB1, DLGAP5, KIF11, and KIF15, showed potential for lung cancer diagnosis and prognosis. In vitro experiments demonstrated that AURKA significantly influenced the proliferation and migration of lung cancer cells by disrupting the cell

cycle. These findings suggest that AURKA may play a crucial role in the occurrence, development, and prognosis of NSCLC [18].

CAT:

CAT was identified as one of the immune-associated genes (IAGs) associated with the prognosis of lung adenocarcinoma (LUAD). A signature comprising CAT and three other IAGs was established, and high-risk scores based on the expression levels of these genes were significantly associated with poor survival outcomes in LUAD patients. CAT's involvement suggests its potential as a prognostic marker for LUAD [19].

CAT was identified as one of the metabolism-related genes (MRGs) used to establish a prognostic signature for LUAD. The prognostic signature, comprising CAT along with five other MRGs (ALDOA, ENTPD2, GNPNAT1, LDHA, TYMS), was validated using LUAD datasets. The signature showed promise as a prognostic tool for LUAD, potentially aiding in diagnosis, individualized therapy, and prognosis [20].

CAT was identified as one of the oxidative stress (OxS)-related genes used to construct a prognostic risk model for LUAD. The risk model, comprising CAT along with three other OxS-related genes (CYP2D6, FMO3, GAPDH), showed good predictive power for LUAD prognosis. High-risk patients exhibited shorter overall survival (OS) and higher tumor mutation burden. CAT overexpression was found to decrease the proliferation, invasion, and migration of lung cancer cells. The risk score based on this model could serve as an independent prognostic factor for LUAD and may aid in individualized immunotherapeutic strategies [21].

CAT was identified as one of the signature genes involved in reactive oxygen species (ROS) regulation and DNA repair in LUAD. Analysis of LUAD transcriptomic data revealed that the expression of ROS-related genes and DNA repair genes had a significant impact on patient survival. The study established a survival prognostic model including CAT along with other genes (TERT, PRKDC, PTTG1, SMUG1, TXNRD1, H2AFX, and PFKP). The risk score derived from this model could serve as an independent prognostic factor in LUAD patients [22].

CAT expression was significantly down-regulated in lung adenocarcinoma (LUAD) tissues compared to normal tissues, and low CAT expression was independently correlated with a worse prognosis in LUAD. CAT down-regulation was associated with an inhibition of the cell cycle. LUAD cases with a p53 mutation exhibited significantly lower CAT expression than those with wild-type p53. CAT expression may serve as a potent favorable prognostic marker for LUAD and could represent a potential drug target [23].

FANCL:

FANCL, a key gene in the Fanconi anemia (FA) pathway, was found to play a crucial role in cisplatin resistance in NSCLC. Knockdown of FANCL significantly increased the sensitivity of cisplatin-resistant NSCLC cells to cisplatin, indicating that FANCL may contribute to acquired cisplatin resistance by enhancing FA pathway capacities responsible for DNA inter-strand crosslink repair[24].

RHOQ:

In lung adenocarcinoma (LUAD), suppressing RhoQ expression promotes TGF-β-mediated Epithelial-to-Mesenchymal Transition (EMT) and invasion in cell lines. RhoQ knockdown increases Smad3 phosphorylation and Snail expression, indicating its involvement in TGF/Smad signaling during the EMT process. Additionally, low RhoQ levels are associated with poor overall survival in LUAD patients [25].

SMC4:

SMC4, a core subunit of condensin complexes, is overexpressed in lung adenocarcinoma tissues and acts as an independent prognostic factor. Knockdown of SMC4 inhibits proliferation and invasion of A549 cells, suggesting its role in lung adenocarcinoma progression [26].

SMC4 is identified as one of the ten core dysregulated genes (DEGs) associated with NSCLC and type 2 diabetes mellitus. The dysregulated immune cells associated with these core DEGs offer a potential avenue for diagnosing and treating lung cancer combined with diabetes [27].

SMC4 is identified as one of the genes present in modules associated with cell cycle progression in lung adenocarcinoma [28].

DSTN:

DSTN is highly expressed in lung adenocarcinoma tissues and is positively correlated with cancer development, metastasis, and poor prognosis in patients. It promotes cell proliferation, invasion, and migration in vitro, as well as tumor formation and lung metastasis in vivo. DSTN facilitates β-catenin nuclear translocation, inducing epithelial-to-mesenchymal transition (EMT), and enhancing lung cancer malignancy. Therefore, DSTN might serve as a therapeutic target and an independent prognostic marker for lung adenocarcinoma [29].

TGFBR2:

TGFBR2 mutation predicts resistance to immune checkpoint inhibitors (ICIs) in NSCLC. Patients with TGFBR2 mutations show significantly shorter progression-free survival (PFS) and overall survival (OS) when treated with ICIs compared to those with wild-type TGFBR2. TGFBR2 mutation is associated with upregulated expression of immune checkpoint-related genes, indicating a link between TGFBR2 mutation and immune resistance in NSCLC [30].

FPR2:

FPR2 was identified as one of the most significantly downregulated genes in NSCLC through analysis of The Cancer Genome Atlas (TCGA) and Gene Expression Omnibus databases. The downregulation of FPR2 in NSCLC suggests its potential role as a biomarker or therapeutic target for the disease [31].

FPR2 was identified as an immune-stemness gene associated with poor prognosis in LUAD. The findings suggest that FPR2 may play a significant role in LUAD development, potentially through cytokine-cytokine receptor interaction and the JAK–STAT pathway [32].

FPR2 was identified as one of the hub genes closely correlated with overall survival time in LUAD patients. The study suggests that FPR2 could serve as a potential prognostic biomarker for LUAD [33].

PLEKHG1:

The gene PLEKHG1 was found to be associated with "dead with disease" outcome in lung adenocarcinoma patients. This suggests its potential role in disease progression and may be used for risk stratification and future treatment development [34].

The gene PLEKHG1 was part of a seven-gene signature associated with the tumor microenvironment (TME) in advanced lung adenocarcinoma (LUAD). This gene signature could serve as a prognosis stratification tool to predict survival outcomes of advanced LUAD patients [35].

YTHDF1:

YTHDF1 is amplified in NSCLC. Its deficiency inhibits NSCLC cell proliferation and tumor formation. Conversely, high YTHDF1 expression correlates with a better clinical outcome, but its depletion can render cancer cells resistant to cisplatin treatment [36].

The m1A modification and its regulators play critical roles in tumorigenesis, including NSCLC. YTHDF1, an m1A reader, is amplified in NSCLC and regulates cancer cell proliferation and response to cisplatin treatment [37].

In LUAD, elevated global m6A levels, resulting from upregulation of METTL3 and downregulation of ALKBH5, are associated with poor patient survival. YTHDF1-mediated mechanisms enhance the translation of enolase 1 (ENO1), stimulating tumorigenesis and glycolysis. Targeting this m6A-dependent pathway, specifically YTHDF1, may offer a potential treatment strategy for LUAD [38].

In NSCLC, METTL3-mediated m6A modification of FRAS1 is associated with poor prognosis. METTL3 regulates FRAS1 m6A modification, and this modification is recognized by YTHDF1.

YTHDF1 cooperates with METTL3 to promote NSCLC cell proliferation, colony formation, and tumor growth through CDON [39].

In NSCLC, YTHDF1, along with other RNA-binding proteins like IGF2BP1/2/3, HuR, and FBL, stabilizes target mRNAs, impacting various pathways such as the JAK-STAT and Hippo signaling pathways, cytokine pathways, cell cycle regulation, and neovascularization. YTHDF1 may also be involved in m6A modification of lncRNAs or target mRNAs [40].

YTHDF1, along with eight other m6A regulators, showed differential expression between TP53-mutant and wild-type NSCLC. ALKBH5 and HNRNPA2B1, in association with YTHDF1, were linked to the prognosis of TP53-mutant NSCLC patients. These findings suggest that m6A regulators, including YTHDF1, could serve as prognostic predictors in TP53-mutant NSCLC [41].

In NSCLC, ALKBH5 inhibits tumor growth and metastasis by reducing YAP expression through YTHDF1-mediated mRNA translation and YTHDF2-mediated mRNA decay. YTHDF1 promotes YAP mRNA translation by interacting with eIF3a, regulated by m6A modification [42].

In NSCLC, YTHDF1 and YTHDF2 expression is associated with favorable prognostic outcomes and increased tumor-infiltrating lymphocytes (TILs). YTHDF1 and YTHDF2 downregulation upregulates PD-L1 expression and alters immune-related gene expression. Thus, YTHDF1 and YTHDF2 could serve as prognostic markers and potential therapeutic targets in NSCLC [43].

YTHDF1 expression is upregulated in NSCLC and correlates with poor clinicopathological features and survival. YTHDF1, along with other m6A regulatory proteins, including METTL3, ALKBH5, and YTHDC2, could serve as predictive markers for NSCLC, aiding in early detection and diagnosis. Additionally, YTHDF1, along with METTL3, ALKBH5, and YTHDC2, is significantly upregulated in NSCLC tissues compared to normal lung tissues [44].

YTHDF1, along with other m6A and m5C regulators, contributes to the crosstalk function in mRNA expression of early-stage LUAD. A seven-gene risk model, including METTL3, NPLOC4, RBM15, YTHDF1, IGF2BP1, NSUN3, and NSUN7, helps stratify the prognosis of early-stage LUAD. High-risk scores are associated with poorer prognosis, indicating the potential of this model as a critical prognostic tool for early-stage LUAD [45].

In NSCLC, YTHDF1 and YTHDF2 expression levels are negatively associated with CD8- and CD4-positive T cells but positively associated with FOXP3-positive T cells. Low expression of YTHDF1 or YTHDF2 is correlated with immune hot tumor gene sets and better prognosis. YTHDF1 and YTHDF2 are predictive markers of response to PD-1/PD-L1 inhibitors, indicating their potential as prognostic markers in NSCLC [46].

In LUAD, YTHDF1 is found to be overexpressed, and its expression is associated with better overall survival (OS) and recurrence-free survival (RFS). This suggests that YTHDF1 could serve as a novel prognostic biomarker for LUAD [47].

In patients with LUAD harboring KRAS/TP53 co-mutations, YTHDF1 is significantly upregulated and associated with poor overall survival. Elevated YTHDF1 promotes the translation of cyclin B1 mRNA in an m6A-dependent manner, facilitating tumor proliferation and leading to an adverse prognosis in LUAD with KRAS/TP53 co-mutations [48].

Aberrant m6A modification was investigated in LUAD. YTHDF1, an m6A reader, was found to be involved in this modification, highlighting its potential prognostic value in LUAD [49].

ELF1:

ELF1 was found to be a promoter of CASC2, a long non-coding RNA (lncRNA) implicated in chemoresistance in non-small cell lung cancer (NSCLC). The ELF1/CASC2/miR-18a axis was identified as a regulatory mechanism affecting the proliferation, migration, and invasion of cisplatin-resistant NSCLC cells, thus affecting patient survival [50].

ELF1 was found to positively regulate miR-152-3p levels by directly interacting with the miR-152-3p promoter. ELF1 inhibited autophagy and reversed cisplatin resistance in NSCLC cells through the miR-152-3p/NCAM1 pathway [51].

ELF1 and survivin expression were positively correlated with intratumoral microvessel density (iMVD) in NSCLC. Their expression levels were significantly related to tumor differentiation, lymphatic metastasis, clinical stage, and postoperative survival time. Additionally, blocking the activity of ELF1 and survivin may offer a new approach to inhibit angiogenesis in NSCLC [52].

ELF1 expression was detected in 72.46% of NSCLC specimens, and its levels were significantly related to tumor differentiation, lymphatic metastasis, clinical stage, and postoperative survival time. High ELF1 expression was correlated with poor prognosis in NSCLC patients. Additionally, ELF1 expression was positively correlated with VEGF expression, suggesting a role for ELF1 in NSCLC progression and angiogenesis [53].

KDM4A:

KDM4A and KDM4D expression was associated with the presence of lymph node metastases in lung carcinomas. Cytoplasmic KDM4A expression correlated with poor patient survival and shorter recurrence-free interval. These findings suggest a significant role for KDM4A and KDM4D in the metastatic spread of lung carcinomas, indicating their involvement in mechanisms associated with cell proliferation, apoptosis, and DNA repair [54].

ATF3:

Loss of CH25H in antigen-presenting cells isolated from human lung tumors is associated with tumor growth and lung cancer progression. This suppression is induced by tumor microenvironment-derived factors that activate the activating transcription factor-3 (ATF3) transcription factor. Downregulation of CH25H stimulates lysosomal degradation, restricts cross-presentation of tumor antigens in intratumoral dendritic cells (DCs), and hinders long-term immunity against malignant cells undergoing chemotherapy-induced immunogenic cell death. These findings suggest that ATF3-mediated downregulation of CH25H in DCs contributes to tumor immune evasion and resistance to therapy in lung cancer [55].

Overexpression of NDRG1 in lung cancer cells reduces cisplatin-induced cytotoxicity by downregulating ATF3 expression. Conversely, overexpression of ATF3 promotes cisplatin-induced cytotoxicity in lung cancer cells. These findings suggest that ATF3 plays a crucial role in regulating cisplatin resistance in lung cancer [56].

CBX8:

CBX8 expression is significantly higher in lung adenocarcinoma (LUAD) tissues compared to adjacent nontumor tissues. It promotes LUAD cell proliferation and migration in vitro. CBX8 directly binds to the promoters of CDKN2C and SCEL, repressing their transcription. Depletion of CDKN2C and SCEL restores the repressed growth and invasion ability of LUAD cells caused by CBX8 knockdown. These findings highlight CBX8's oncogenic role in LUAD progression [57].

CEMIP:

ALKBH5 downregulation in paclitaxel (PTX) resistant NSCLC cells correlates with poor prognosis in NSCLC patients. It modulates PTX sensitivity and epithelial-mesenchymal transition (EMT) by regulating CEMIP expression. ALKBH5 reduces CEMIP mRNA stability, implicating the ALKBH5/CEMIP axis in NSCLC chemoresistance [58].

RUVBL1: (6)

RUVBL1/2 ATPase activity is overexpressed in NSCLC tumors, correlating with poor survival. Inhibition of RUVBL1/2 ATPase activity induces S-phase arrest, leading to cancer cell death via replication catastrophe. Additionally, RUVBL1/2 inhibition synergizes with radiation therapy in NSCLC, offering a potential therapeutic strategy [59].

High levels of RUVBL1 and HNRNPU proteins and mRNA are associated with poor overall survival (OS) in stage I and II NSCLC patients. Co-expression of RUVBL1 and HNRNPU (R + H +) further exacerbates poor prognosis, suggesting their potential as prognostic biomarkers and therapeutic targets for NSCLC [60].

RUVBL1 has been identified as a contributor to TRAIL resistance in NSCLC cells by repressing c-Jun/AP-1 activity. Knocking down RUVBL1 sensitizes resistant cells to TRAIL-induced apoptosis, while its overexpression inhibits TRAIL-induced cell death. High RUVBL1 expression, inversely correlated with low c-Jun levels, is associated with poor overall prognosis in LUAD. These findings suggest that targeting RUVBL1 in combination with TRAIL may offer a novel therapeutic strategy for lung cancer treatment [61].

RUVBL1 is overexpressed in lung adenocarcinoma tissues and cell lines. Knocking down RUVBL1 inhibits lung adenocarcinoma cell proliferation by inducing G1/S phase cell cycle arrest through multiple mechanisms. This suggests RUVBL1 as a potential therapeutic target for lung adenocarcinoma [62].

RUVBL1 is identified as one of the genes with concordant changes in DNA copy number and expression levels in non-small cell lung cancer (NSCLC). It is overexpressed and located in an amplified region, suggesting its potential role in lung cancer development and progression [63].

RUVBL1 activates the RAF/MEK/ERK pathway by inhibiting phosphorylation of C-RAF at serine 259, promoting lung cancer progression. Its elevated expression in lung adenocarcinoma tissues suggests its potential as a therapeutic target for lung cancer treatment [64].

NUP155:

NUP155 expression is higher in grade 3 lung adenocarcinoma, suggesting its potential involvement in tumor grading and chemoresistance [65].

Supplementary Material S1 Bibliography

[1] J. Liu *et al.*, “Immune microenvironment analysis and novel biomarkers of early-stage lung adenocarcinoma evolution,” *Front. Oncol.*, vol. 13, no. June, pp. 1–10, 2023, doi: 10.3389/fonc.2023.1150098.

[2] K. Liu, L. Cheng, K. Zhu, J. Wang, and Q. Shu, “The cancer/testis antigen HORMAD1 mediates epithelial–mesenchymal transition to promote tumor growth and metastasis by activating the Wnt/β-catenin signaling pathway in lung cancer,” *Cell Death Discov.*, vol. 8, no. 1, 2022, doi: 10.1038/s41420-022-00946-1.

[3] Diering and J. F. Nishijima, Daniel; K. Simel, David L; Wisner, David H; Holmes, “HORMAD1 Is a Negative Prognostic Indicator in Lung Adenocarcinoma and Specifies Resistance to Oxidative and Genotoxic Stress,” *Physiol. Behav.*, vol. 176, no. 1, pp. 139–148, 2016, doi: 10.1158/0008-5472.CAN-18-1377.HORMAD1.

[4] J. Huang *et al.*, “BDH1-mediated LRRC31 regulation dependent on histone lysine β-hydroxybutyrylation to promote lung adenocarcinoma progression,” *MedComm*, vol. 4, no. 6, pp. 1–18, 2023, doi: 10.1002/mco2.449.

[5] J. Zhang *et al.*, “A novel iTreg-related signature for prognostic prediction in lung adenocarcinoma,” *Cancer Sci.*, vol. 115, no. 1, pp. 109–124, 2024, doi: 10.1111/cas.16015.

[6] X. Hu *et al.*, “Serum levels of retinol-binding protein 4 and the risk of non-small cell lung cancer: A case-control study,” *Med. (United States)*, vol. 99, no. 31, p. E21254, 2020, doi: 10.1097/MD.0000000000021254.

[7] F. Ávila-Moreno *et al.*, “Overexpression of MEOX2 and TWIST1 is associated with H3K27me3 levels and determines lung cancer chemoresistance and prognosis,” *PLoS One*, vol. 9, no. 12, 2014, doi: 10.1371/journal.pone.0114104.

[8] L. Armas-López *et al.*, “Epigenomic study identifies a novel mesenchyme homeobox 2-GLI1 transcription axis involved in cancer drug resistance overall survival and therapy prognosis in lung cancer patients,” *Oncotarget*, vol. 8, no. 40, pp. 67056–67081, 2017, doi: 10.18632/oncotarget.17715.

[9] X. Ji, R. Zhu, C. Gao, H. Xie, X. Gong, and J. Luo, “Hypoxia-Derived Exosomes Promote Lung Adenocarcinoma by Regulating HS3ST1-GPC4-Mediated Glycolysis,” *Cancers (Basel).*, vol. 16, no. 4, 2024, doi: 10.3390/cancers16040695.

[10] J. Wang *et al.*, “Repression of the AURKA-CXCL5 axis induces autophagic cell death and promotes radiosensitivity in non-small-cell lung cancer,” *Cancer Lett.*, vol. 509, no. March, pp. 89–104, 2021, doi: 10.1016/j.canlet.2021.03.028.

[11] L. Wang *et al.*, “Identification and validation of key genes with prognostic value in non-small-cell lung cancer via integrated bioinformatics analysis,” *Thorac. Cancer*, vol. 11, no. 4, pp. 851–866, 2020, doi: 10.1111/1759-7714.13298.

[12] M. A. Schneider *et al.*, “AURKA, DLGAP5, TPX2, KIF11 and CKAP5: Five specific mitosis-associated genes correlate with poor prognosis for non-small cell lung cancer patients,” *Int. J. Oncol.*, vol. 50, no. 2, pp. 365–372, 2017, doi: 10.3892/ijo.2017.3834.

[13] A. S. K. Al-Khafaji *et al.*, “AURKA mRNA expression is an independent predictor of poor prognosis in patients with non-small cell lung cancer,” *Oncol. Lett.*, vol. 13, no. 6, pp. 4463–4468, 2017, doi: 10.3892/ol.2017.6012.

[14] N. Liu *et al.*, “Inhibition of Aurora A enhances radiosensitivity in selected lung cancer cell lines,” *Respir. Res.*, vol. 20, no. 1, pp. 1–15, 2019, doi: 10.1186/s12931-019-1194-8.

[15] M. Peindl *et al.*, “EMT, Stemness, and Drug Resistance in Biological Context: A 3D Tumor Tissue/In Silico Platform for Analysis of Combinatorial Treatment in NSCLC with Aggressive KRAS-Biomarker Signatures,” *Cancers (Basel).*, vol. 14, no. 9, 2022, doi: 10.3390/cancers14092176.

[16] J. Chen *et al.*, “Non-small-cell lung cancer pathological subtype-related gene selection and bioinformatics analysis based on gene expression profiles,” *Mol. Clin. Oncol.*, pp. 356–361, 2017, doi: 10.3892/mco.2017.1516.

[17] Ö. C. Erkin, B. Cömertpay, and E. Göv, “Integrative Analysis for Identification of Therapeutic Targets and Prognostic Signatures in Non-Small Cell Lung Cancer,” *Bioinform. Biol. Insights*, vol. 16, 2022, doi: 10.1177/11779322221088796.

[18] S. Shi, Y. Qiu, Z. Jin, J. Zhou, W. Yu, and H. Zhang, “AURKA Identified as Potential Lung Cancer Marker through Comprehensive Bioinformatic Analysis and Experimental Verification.,” *Crit. Rev. Eukaryot. Gene Expr.*, vol. 33, no. 5, pp. 39–59, 2023, doi: 10.1615/CritRevEukaryotGeneExpr.2023046830.

[19] Z. Wang and X. Chen, “Establishment and validation of an immune-associated signature in lung adenocarcinoma,” *Int. Immunopharmacol.*, vol. 88, no. July, p. 106867, 2020, doi: 10.1016/j.intimp.2020.106867.

[20] Z. Wang *et al.*, “Establishment and validation of a prognostic signature for lung adenocarcinoma based on metabolism-related genes,” *Cancer Cell Int.*, vol. 21, no. 1, pp. 1–16, 2021, doi: 10.1186/s12935-021-01915-x.

[21] Y. Zhu *et al.*, “Identification of a novel oxidative stress-related prognostic model in lung adenocarcinoma,” *Front. Pharmacol.*, vol. 13, no. November, pp. 1–15, 2022, doi: 10.3389/fphar.2022.1030062.

[22] Y. Zhao, H. M. Feng, W. J. Yan, and Y. Qin, “Identification of the Signature Genes and Network of Reactive Oxygen Species Related Genes and DNA Repair Genes in Lung Adenocarcinoma,” *Front. Med.*, vol. 9, no. February, pp. 1–14, 2022, doi: 10.3389/fmed.2022.833829.

[23] P.-M. CHEN, Y.-H. HUANG, H.-H. CHEN, and P.-Y. CHU, “Catalase Expression Is an Independent Prognostic Marker in Lung Adenocarcinoma,” *Anticancer Res.*, vol. 44, no. 1, pp. 287–300, 2024, doi: 10.21873/anticanres.16811.

[24] P. Chen *et al.*, “The functional status of DNA repair pathways determines the sensitization effect to cisplatin in non-small cell lung cancer cells,” *Cell. Oncol.*, vol. 39, no. 6, pp. 511–522, 2016, doi: 10.1007/s13402-016-0291-7.

[25] K. Satoh, S. Sakai, and M. Nishizuka, “Knockdown of RhoQ, a member of Rho GTPase, accelerates TGF-β-induced EMT in human lung adenocarcinoma,” *Biochem. Biophys. Reports*, vol. 32, no. August, p. 101346, 2022, doi: 10.1016/j.bbrep.2022.101346.

[26] C. Zhang, M. Kuang, M. Li, L. Feng, K. Zhang, and S. Cheng, “SMC4, which is essentially involved in lung development, is associated with lung adenocarcinoma progression,” *Sci. Rep.*, vol. 6, no. May, pp. 1–11, 2016, doi: 10.1038/srep34508.

[27] Q. Yuan, L. Li, L. S. Wang, and S. G. Xing, “Epidemiological and transcriptome data identify shared gene signatures and immune cell infiltration in type 2 diabetes and non-small cell lung cancer,” *Diabetol. Metab. Syndr.*, vol. 16, no. 1, pp. 1–20, 2024, doi: 10.1186/s13098-024-01278-z.

[28] G. Bidkhori, Z. Narimani, S. Hosseini Ashtiani, A. Moeini, A. Nowzari-Dalini, and A. Masoudi-Nejad, “Reconstruction of an Integrated Genome-Scale Co-Expression Network Reveals Key Modules Involved in Lung Adenocarcinoma,” *PLoS One*, vol. 8, no. 7, pp. 1–10, 2013, doi: 10.1371/journal.pone.0067552.

[29] H. J. Zhang *et al.*, “Destrin contributes to lung adenocarcinoma progression by activating Wnt/β-catenin signaling pathway,” *Mol. Cancer Res.*, vol. 18, no. 12, pp. 1789–1802, 2020, doi: 10.1158/1541-7786.MCR-20-0187.

[30] R. L. Soiza, A. I. C. Donaldson, and P. K. Myint, “TGFBR2 mutation predicts resistance to immune checkpoint inhibitors in patients with non-small cell lung cancer,” *Ther. Adv. Vaccines*, vol. 9, no. 6, pp. 259–261, 2018, doi: 10.1177/https.

[31] N. Kang, W.-J. Qiu, B. Wang, D.-F. Tang, and X.-Y. Shen, “Role of hemoglobin alpha and hemoglobin beta in non-small-cell lung cancer based on bioinformatics analysis.,” *Mol. Carcinog.*, vol. 61, no. 6, pp. 587–602, Jun. 2022, doi: 10.1002/mc.23404.

[32] H. Wang *et al.*, “Integrative stemness characteristics associated with prognosis and the immune microenvironment in lung adenocarcinoma,” *BMC Pulm. Med.*, vol. 22, no. 1, pp. 1–11, 2022, doi: 10.1186/s12890-022-02184-8.

[33] Y. Yu and X. Tian, “Analysis of genes associated with prognosis of lung adenocarcinoma based on GEO and TCGA databases,” *Med. (United States)*, vol. 99, no. 19, p. E20183, 2020, doi: 10.1097/MD.0000000000020183.

[34] F. Deng, L. Shen, H. Wang, and L. Zhang, “Classify multicategory outcome in patients with lung adenocarcinoma using clinical, transcriptomic and clinico-transcriptomic data: machine learning versus multinomial models.,” *Am. J. Cancer Res.*, vol. 10, no. 12, pp. 4624–4639, 2020, [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/33415023%0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC7783755

[35] H. Zhao, X. Zhang, L. Guo, S. Shi, and C. Lu, “A Robust Seven-Gene Signature Associated With Tumor Microenvironment to Predict Survival Outcomes of Patients With Stage III–IV Lung Adenocarcinoma,” *Front. Genet.*, vol. 12, no. September, pp. 1–13, 2021, doi: 10.3389/fgene.2021.684281.

[36] Y. Shi *et al.*, “YTHDF1 links hypoxia adaptation and non-small cell lung cancer progression,” *Nat. Commun.*, vol. 10, no. 1, 2019, doi: 10.1038/s41467-019-12801-6.

[37] J. Li, H. Zhang, and H. Wang, “N1-methyladenosine modification in cancer biology: Current status and future perspectives,” *Comput. Struct. Biotechnol. J.*, vol. 20, pp. 6578–6585, 2022, doi: 10.1016/j.csbj.2022.11.045.

[38] L. Ma *et al.*, “The essential roles of m6A RNA modification to stimulate ENO1-dependent glycolysis and tumorigenesis in lung adenocarcinoma,” *J. Exp. Clin. Cancer Res.*, vol. 41, no. 1, pp. 1–21, 2022, doi: 10.1186/s13046-021-02200-5.

[39] X. Dou, Z. Wang, W. Lu, L. Miao, and Y. Zhao, “METTL3 promotes non-small cell lung cancer (NSCLC) cell proliferation and colony formation in a m6A-YTHDF1 dependent way,” *BMC Pulm. Med.*, vol. 22, no. 1, pp. 1–15, 2022, doi: 10.1186/s12890-022-02119-3.

[40] E. A. Braga *et al.*, “Various LncRNA Mechanisms in Gene Regulation Involving miRNAs or RNA-Binding Proteins in Non-Small-Cell Lung Cancer: Main Signaling Pathways and Networks,” *Int. J. Mol. Sci.*, vol. 24, no. 17, 2023, doi: 10.3390/ijms241713617.

[41] Z. Zhao *et al.*, “Expression and prognostic significance of m6A-related genes in TP53-mutant non-small-cell lung cancer,” *J. Clin. Lab. Anal.*, vol. 36, no. 1, pp. 1–11, 2022, doi: 10.1002/jcla.24118.

[42] D. Jin *et al.*, “M6A demethylase ALKBH5 inhibits tumor growth and metastasis by reducing YTHDFs-mediated YAP expression and inhibiting miR-107/LATS2-mediated YAP activity in NSCLC,” *Mol. Cancer*, vol. 19, no. 1, pp. 1–24, 2020, doi: 10.1186/s12943-020-01161-1.

[43] K. Tsuchiya *et al.*, “YTHDF1 and YTHDF2 are associated with better patient survival and an inflamed tumor-immune microenvironment in non–small-cell lung cancer,” *Oncoimmunology*, vol. 10, no. 1, pp. 1–13, 2021, doi: 10.1080/2162402X.2021.1962656.

[44] Y. Li *et al.*, “The pathological tissue expression pattern and clinical significance of m6A-regulatory genes in non-small cell lung cancer.,” *J. Gene Med.*, vol. 24, no. 2, p. e3397, Feb. 2022, doi: 10.1002/jgm.3397.

[45] L. Tian, Y. Wang, J. Tian, W. Song, L. Li, and G. Che, “Prognostic Value and Genome Signature of m6A/m5C Regulated Genes in Early-Stage Lung Adenocarcinoma,” *Int. J. Mol. Sci.*, vol. 24, no. 7, pp. 1–19, 2023, doi: 10.3390/ijms24076520.

[46] Y. W. Koh, J. H. Han, S. Haam, and H. W. Lee, “Prognostic and predictive value of YTHDF1 and YTHDF2 and their correlation with tumor-infiltrating immune cells in non-small cell carcinoma,” *Front. Oncol.*, vol. 12, no. November, pp. 1–12, 2022, doi: 10.3389/fonc.2022.996634.

[47] Y. Zhang, X. Liu, L. Liu, J. Li, Q. Hu, and R. Sun, “Expression and prognostic significance of m6A-related genes in lung adenocarcinoma,” *Med. Sci. Monit.*, vol. 26, 2020, doi: 10.12659/MSM.919644.

[48] X. Lou *et al.*, “Ythdf1 promotes cyclin b1 translation through m6 a modulation and contributes to the poor prognosis of lung adenocarcinoma with kras/tp53 co-mutation,” *Cells*, vol. 10, no. 7, pp. 1–13, 2021, doi: 10.3390/cells10071669.

[49] J. Liu, Z. Zheng, and J. Zhong, “Function and prognostic value of N6-methyladenosine-modified RNAs in lung adenocarcinoma.,” *J. Gene Med.*, vol. 25, no. 1, p. e3454, Jan. 2023, doi: 10.1002/jgm.3454.

[50] X. H. Xiao and S. Y. He, “ELF1 activated long non-coding RNA CASC2 inhibits cisplatin resistance of non-small cell lung cancer via the miR-18a/IRF-2 signaling pathway,” *Eur. Rev.*

Med. Pharmacol. Sci., vol. 24, no. 6, pp. 3130–3142, 2020, doi: 10.26355/eurrev_202003_20680.

[51] L. Zhao, X. Wu, Z. Zhang, L. Fang, B. Yang, and Y. Li, "ELF1 suppresses autophagy to reduce cisplatin resistance via the miR-152-3p/NCAM1/ERK axis in lung cancer cells," *Cancer Sci.*, vol. 114, no. 6, pp. 2650–2663, 2023, doi: 10.1111/cas.15770.

[52] D.-X. Yang, N.-E. Li, Y. Ma, Y.-C. Han, and Y. Shi, "Expression of Elf-1 and survivin in non-small cell lung cancer and their relationship to intratumoral microvessel density.," *Chin. J. Cancer*, vol. 29, no. 4, pp. 396–402, Apr. 2010, doi: 10.5732/cjc.009.10547.

[53] D.-X. Yang, Y.-C. Han, L.-Y. Liu, N. Yu, X. Wang, and Y. Shi, "[Expression and significance of Elf-1 and vascular endothelial growth factor in non-small cell lung cancer].," *Ai Zheng*, vol. 28, no. 7, pp. 762–767, Jul. 2009, doi: 10.5732/cjc.008.10748.

[54] Y. Soini, V.-M. Kosma, and R. Pirinen, "KDM4A, KDM4B and KDM4C in non-small cell lung cancer.," *Int. J. Clin. Exp. Pathol.*, vol. 8, no. 10, pp. 12922–12928, 2015.

[55] Z. Lu *et al.*, "Tumor factors stimulate lysosomal degradation of tumor antigens and undermine their cross-presentation in lung cancer," *Nat. Commun.*, vol. 13, no. 1, pp. 1–17, 2022, doi: 10.1038/s41467-022-34428-w.

[56] A. Du, Y. Jiang, and C. Fan, "NDRG1 downregulates ATF3 and inhibits cisplatininduced cytotoxicity in lung cancer A549 cells," *Int. J. Med. Sci.*, vol. 15, no. 13, pp. 1502–1507, 2018, doi: 10.7150/ijms.28055.

[57] H. Chen *et al.*, "CBX8 promotes lung adenocarcinoma growth and metastasis through transcriptional repression of CDKN2C and SCEL.," *J. Cell. Physiol.*, vol. 238, no. 11, pp. 2710–2723, Nov. 2023, doi: 10.1002/jcp.31124.

[58] L. Gao *et al.*, "ALKBH5 regulates paclitaxel resistance in NSCLC via inhibiting CEMIP-mediated EMT," *Toxicol. Appl. Pharmacol.*, vol. 483, no. December 2023, p. 116807, 2024, doi: 10.1016/j.taap.2024.116807.

[59] P. Yenerall *et al.*, "RUVBL1/RUVBL2 ATPase Activity Drives PAQosome Maturation, DNA Replication and Radioresistance in Lung Cancer," vol. 27, no. 1, pp. 105–121, 2021, doi: 10.1016/j.chembiol.2019.12.005.RUVBL1/RUVBL2.

[60] J. Durślewicz *et al.*, "High expression of RUVBL1 and HNRNPU is associated with poor overall survival in stage I and II non-small cell lung cancer patients," *Discov. Oncol.*, vol. 13, no. 1, 2022, doi: 10.1007/s12672-022-00568-0.

[61] H. Li, T. Zhou, Y. Zhang, H. Jiang, J. Zhang, and Z. Hua, "RuvBL1 Maintains Resistance to TRAIL-Induced Apoptosis by Suppressing c-Jun/AP-1 Activity in Non-Small Cell Lung Cancer," *Front. Oncol.*, vol. 11, no. June, pp. 1–8, 2021, doi: 10.3389/fonc.2021.679243.

[62] X.-S. Yuan *et al.*, "Downregulation of RUVBL1 inhibits proliferation of lung adenocarcinoma cells by G1/S phase cell cycle arrest via multiple mechanisms.," *Tumour*

Biol. J. Int. Soc. Oncodevelopmental Biol. Med., Oct. 2016, doi: 10.1007/s13277-016-5452-9.

[63] E. Dehan *et al.*, “Chromosomal aberrations and gene expression profiles in non-small cell lung cancer.,” *Lung Cancer*, vol. 56, no. 2, pp. 175–184, May 2007, doi: 10.1016/j.lungcan.2006.12.010.

[64] H. Guo *et al.*, “RUVBL1, a novel C-RAF-binding protein, activates the RAF/MEK/ERK pathway to promote lung cancer tumorigenesis,” *Biochem. Biophys. Res. Commun.*, vol. 498, no. 4, pp. 932–939, 2018, doi: 10.1016/j.bbrc.2018.03.084.

[65] F. Forest *et al.*, “WHO grading system for invasive pulmonary lung adenocarcinoma reveals distinct molecular signature: An analysis from the cancer genome atlas database,” *Exp. Mol. Pathol.*, vol. 125, no. March, p. 104756, 2022, doi: 10.1016/j.yexmp.2022.104756.

APPENDIX: Formulas and Notations

Bayesian Network:

Definition: A Bayesian Network is a probabilistic graphical model that represents a set of variables and their conditional dependencies via a directed acyclic graph (DAG). Each node in the graph represents a variable, and the edges represent the conditional dependencies between these variables.

Formula:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid Parents(X_i))$$

where n is the number of variables and $P(X_i|Parents(X_i))$ is the conditional probability of X_i given its parents in the graph.

Key characteristics:

- Models the joint distribution of a set of variables.
- Useful for inferring causal relationships and predicting the effects of interventions.
- Can handle both discrete and continuous variables.

Applications: Gene network inference to model causal relationships between genes, risk assessment, diagnostic systems and decision support.

Context: Section 1.2.1 Bayesian networks.

Binomial Distribution:

Definition: The binomial distribution is a discrete probability distribution that describes the number of successes in a fixed number of independent Bernoulli trials, each with the same probability of success. It is denoted as $B(n, p)$, where n is the number of trials, and p is the probability of success in each trial.

Probability Mass Function (PMF): The probability of obtaining exactly k successes in n trials is given by:

$$P(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$

where $\binom{n}{k}$ is the binomial coefficient, calculated as $\frac{n!}{k!(n-k)!}$.

Context: Section 3.1.1 Generation of Synthetic Dataset.

Fisher's Exact Test:

Definition: Fisher's Exact Test is a statistical significance test used to determine if there are nonrandom associations between two categorical variables. It is particularly useful for small sample sizes and 2x2 contingency tables.

Purpose: It tests the null hypothesis that there is no association between the two variables, meaning that the proportions of one variable are independent of the levels of the other variable.

Contingency Table: The test uses a 2x2 contingency table to display the frequencies of the variables.

	In Variable B = 1	**In Variable B = 2**
Variable A = 1	a	b
Variable A = 0	c	d

where

- **a:** Number of times both A = 1 and B = 1
- **b:** Number of times A = 1 and B = 0
- **c:** Number of times A = 0 and B = 1
- **d:** Number of times both A = 0 and B = 0

Formula:

$$\text{p-value} = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!}$$

where $n = a+b+c+d$ is the total number of genes considered.

Context: Section 3.2.1 Validation with Over-representation analysis.

Fused Sparse Structural Equation Modeling (FSSEM):

Definition: The Fused Sparse Structural Equation Modeling (SEM) algorithm (Zhou et al., 2020) is designed to infer Gene Regulatory Networks (GRNs) across two different conditions simultaneously. It utilizes SEM with all observable variables of gene expression and gene perturbations, solving the inference problem by employing an adaptive generalized fused LASSO regression model.

Formula:

$$y_i^{(k)} = B^{(k)} y_i^{(k)} + F^{(k)} x_i^{(k)} + \mu_i^{(k)} + \epsilon_i^{(k)}$$

where

$k=1,2$ are the two different conditions considered by the model

$i=1,\dots,n_k$ is the index of the considered gene for each condition $k=1,2$

$B^{(k)}=[B^{(1)},B^{(2)}]$ is a $n x n$ matrix of p genes representing the unkown network structure under condition k

$F^{(k)}=[F^{(1)},F^{(2)}]$ is a $n x q$ matrix of p genes and q cis-eQTLs that captures the effect of of cis-eQTLs on gene expression.

$\mu_i^{(k)}$ is a $n x 1$ vector that accounts for the model bias in the SEM

$\epsilon_i^{(k)}$ is a $n x 1$ the vector the Gaussian noise with mean zero and variance σ^2

Context: Section 2.3 Fused Sparse Structural Equation Modeling (FSSEM).

Gaussian Graphical Models (GGMs):

Definition: GGMs are statistical models that represent the conditional independence structure between multiple Gaussian-distributed variables using a graph.

Key Concepts:

- Graph Representation: Nodes represent variables, and edges represent conditional dependencies between the variables.
- Precision Matrix: The inverse of the covariance matrix ($\Theta = \Sigma^{-1}$) is used to determine the edges in the graph. An edge between nodes i and j exists if and only if $\Theta_{ij} \neq 0$.

Formula:

$$p(x) = \frac{1}{(2\pi)^{k/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

where Σ is the covariance matrix and μ is the mean vector

Key characteristics:

- Provides a way to visualize and analyze the conditional dependencies between variables.
- Particularly useful for high-dimensional data (i.e. gene expression, finance, etc).

Context: Section 1.2.1 correlation networks.

Gene Regulatory Network (GRN):

Definition: A Gene Regulatory Network (GRN) is a representation of the regulatory interactions between genes within a cell. It is modeled as a graph $G = (V, E)$, where V represents the set of genes (vertices or nodes), and E represents the regulatory relationships (edges or links) between these genes.

- Vertices (Nodes): Represent individual genes.
- Edges (Links): Represent the regulatory influences or interactions that one gene exerts on another.

Types of Relationships:

- Activation (positive regulation): One gene increases the expression of another gene.
- Repression (negative regulation): One gene decreases the expression of another gene.

Graph:

Definition: A graph is a mathematical structure used to model pairwise relationships between objects. It consists of a set of vertices (nodes) and a set of edges (links) that connect pairs of vertices. A graph is typically denoted as $G=(V,E)$, where V represents the set of vertices and E represents the set of edges. Vertices (nodes) are the fundamental units or points in a graph, and edges (links) are the connections between pairs of vertices.

Directed Graph: A graph in which edges have a direction, indicating a one-way relationship from one vertex to another.
Undirected Graph: A graph in which edges have no direction, indicating a mutual relationship between vertices.
Directed Acyclic Graph (DAG): A type of directed graph with no directed cycles, meaning there is no way to start at any vertex and follow a consistently directed path that eventually loops back to the starting vertex.
Directed Cyclic Graph (DCG): A type of directed graph that contains at least one directed cycle, meaning there exists a path where one can start at a vertex, follow the directed edges, and return to the starting vertex.

Maximum Likelihood Estimation (MLE):

Definition: Maximum Likelihood Estimation is a statistical method used to estimate the parameters of a statistical model. It aims to find the parameter values that maximize the likelihood function, which represents the probability of observing the given data under the assumed model.

Key Concepts:

- Likelihood Function: The function that describes the probability of observing the data given the parameter values of the model.

- Log-Likelihood: The natural logarithm of the likelihood function, often used for computational convenience and numerical stability.

Procedure:

- Specify the Model: Define the probability distribution that best describes the data.
- Formulate the Likelihood Function: Express the probability of observing the data as a function of the model parameters.
- Maximize the Likelihood: Find the parameter values that maximize the likelihood function, typically by taking derivatives and solving for the critical points.
- Interpretation: The estimated parameter values represent the most likely values given the observed data and the assumed model.

Context: Solution method for Structural Equation Model (SEM), section 2.1.

Multivariate Normal Distribution:

Definition: The multivariate normal distribution is a generalization of the normal distribution to multiple variables. It describes a set of d variables that are jointly normally distributed. Each variable has a normal distribution, and any linear combination of the variables also follows a normal distribution.

Parameters:

- Mean Vector (μ): A d-dimensional vector representing the means of each variable.
- Covariance Matrix (Σ) A d×d symmetric, positive-definite matrix representing the covariances between the variables.

Probability Density Function (PDF):

The probability density function for a d-dimensional multivariate normal distribution is given by:

$$f(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

where x is a d-dimensional random vector, μ is the mean vector, Σ is the covariance matrix, $|\Sigma|$ is the determinant of Σ, and Σ^{-1} is the inverse of Σ.

Context: Section 3.1.1 Generation of Synthetic Dataset.

Mutual Information:

Definition: Mutual Information (MI) measures the amount of information obtained about one random variable through another random variable. It quantifies the dependency between the variables without assuming any specific type of relationship (linear or monotonic).

Formula:

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

where

- $p(x,y)$: is the joint probability distribution of X and Y
- $p(x)$ and $p(y)$ are the marginal probability distributions of X and Y, respectively.

Key Characteristics:

- Type of relationship: measures any kind of dependency, not restricted to linear or monotonic relationships
- Range: Non-negative values, $I(X;Y) \geq 0$
  - $I(X;Y) = 0$: indicates X and Y are independent
  - Higher values indicate a higher degree of dependency
- Data requirements: can be used with any type if data (continuous, discrete, ordinal)
- Sensitivity to outliers: less sensitive to outliers compared to Pearson correlation.

Context: Section 1.2.1 similarity measure for relevance networks.

Ordinary Differential Equations (ODEs):

Definition: Ordinary Differential Equations are mathematical equations that describe the rate of change of a function with respect to one independent variable. They are used to model systems where the rate of change of a quantity depends only on its current value and not on its history.

Key Concepts:

- Dependent Variable: The function or quantity being modeled.
- Independent Variable: The variable with respect to which the function changes.
- Order: The highest derivative present in the equation. For example, a first-order ODE involves only first derivatives.

$$\text{First-Order ODE:} \quad \frac{dy}{dx} = f(x, y)$$

$$\text{Second-Order ODE:} \quad \frac{d^2y}{dx^2} = f\left(x, y, \frac{dy}{dx}\right)$$

Solution Methods:

- Analytical Methods: Exact solutions obtained through integration techniques, often feasible for simple ODEs.
- Numerical Methods: Approximate solutions obtained through numerical techniques such as Euler's method, Runge-Kutta methods, and finite difference methods.

Context: Section 1.2.1 Differential equation models.

Partial correlation:

Definition: Partial correlation measures the degree of association between two random variables, with the effect of a set of controlling random variables removed.

Formula:

$$r_{xy \cdot z} = \frac{r_{xy} - r_{xz} r_{yz}}{\sqrt{(1 - r_{xz}^2)(1 - r_{yz}^2)}}$$

where r_{xy} is the correlation between x and y, r_{xz} is the correlation between x and z, and r_{yz} is the correlation between y and z.

Key characteristics:

- Assesses direct relationship between two variables by removing the influence of other variables.
- Useful in understanding the unique contribution of variables in multivariate settings.

Context: Section 1.2.1 similarity measure for correlation networks.

Pearson Correlation:

Definition: The Pearson correlation coefficient (denoted as *r*) measures the linear relationship between two continuous variables. It assumes that the relationship between the variables is linear and that the variables are approximately normally distributed.

Formula:

$$r = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$$

where

x_i : individual sample point for variable x

$\bar{x}$: mean of variable x

y_i : individual sample point for variable y

$\bar{y}$: mean of variable y

Key characteristics:

- Linear relationship: Measures the strength and direction of linear relationships between two variables.
- Range: Values range from -1 to 1
  - $r = 1$: perfect positive linear correlation
  - $r = -1$: perfect negative linear correlation
  - $r = 0$: no linear correlation
- Sensitive to outliers: Pearson correlation is sensitive to outliers, which can significantly affect the correlation coefficient.

Context: Section 1.2.1 similarity measure for correlation networks.

Penalized Negative Log-Likelihood:

Definition: The Penalized Negative Log-Likelihood is an extension of maximum likelihood estimation (MLE) used for parameter estimation in models with regularization. It combines the negative log-likelihood function with a penalty term that penalizes complex models to prevent overfitting.

Key Concepts:

- Negative Log-Likelihood (NLL): The negative of the logarithm of the likelihood function, representing the measure of how well the model fits the observed data.
- Penalty Term: An additional term added to the NLL function to penalize complexity, typically based on the parameters or their magnitudes.
- Objective Function:

$$Penalized\, negative \log - likelihood \;=\; -\log - likelihood \;+\; penalty\, term$$

Penalty Types:

- L1 Penalty (Lasso): Penalizes the absolute values of the coefficients, encouraging sparsity and feature selection.
- L2 Penalty (Ridge): Penalizes the squared magnitudes of the coefficients, discouraging large coefficients and reducing multicollinearity.

- Elastic Net Penalty: Combines both L1 and L2 penalties to leverage the benefits of both regularization methods.

Optimization:

- Optimization Algorithms: Various optimization algorithms such as gradient descent, coordinate descent, or proximal gradient methods are used to minimize the penalized NLL function.
- Cross-Validation: Often used to select the optimal regularization parameter(s) and assess the model's predictive performance.

Applications:

- Regression Analysis: Regularizing regression models such as linear regression, logistic regression, or Poisson regression.
- Machine Learning: Regularizing machine learning models like support vector machines (SVM), neural networks, or random forests.
- High-Dimensional Data: Handling datasets with a large number of predictors relative to the sample size, reducing overfitting and improving model generalization.

Context: Section 2.3.1 Joint inference of GRN in FSSEM.

Performance Metrics:

Definitions:

- True Positives (TP): The number of instances correctly identified as positive.
- False Positives (FP): The number of instances incorrectly identified as positive.
- True Negatives (TN): The number of instances correctly identified as negative.
- False Negatives (FN): The number of instances incorrectly identified as negative.

Precision: Precision, also known as positive predictive value, is the ratio of true positives to the total number of predicted positives. It measures the accuracy of the positive predictions.

$$Precision = \frac{TP}{TP+FP}$$

Interpretation: High precision indicates that the model has a low false positive rate.

Recall: Recall, also known as sensitivity or true positive rate, is the ratio of true positives to the total number of actual positives. It measures the ability of the model to identify all relevant instances.

$$Recall = \frac{TP}{TP+FN}$$

Interpretation: High recall indicates that the model has a low false negative rate.

Accuracy: Accuracy is the ratio of correctly predicted instances (both true positives and true negatives) to the total number of instances. It measures the overall correctness of the model.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Interpretation: High accuracy indicates that the model makes a large proportion of correct predictions.

Context: Section 3.1.3 Classification and performance metrics.

Regression methods:

Definition: Regression methods are statistical techniques used to model and analyze the relationships between a dependent variable (response) and one or more independent variables (predictors). The primary goal is to understand the nature of these relationships and make predictions.

Key concepts:

- Dependent variable (Y): The outcome or variable being predicted or explained
- Independent Variables (X): The variables used to predict or explain the dependent variable
- Regression Coefficients: Parameters that quantify the relationship between the independent variables and the dependent variable.

Common types:

- **Linear Regression:** Models the relationship using a linear equation.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + + \beta_n X_n + \epsilon$$

  where β_0 is the intercept, β_i are the coefficients, and ϵ is the error term.

- **Multiple Linear Regression:** Extends linear regression to multiple predictors.
- **Logistic Regression:** Used for binary outcomes, modeling the probability of the dependent variable.

$$\log\left(\frac{P(Y=1)}{P(Y=0)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + + \beta_n X_n + \epsilon$$

- **Polynomial Regression:** Models the relationship using a polynomial equation to capture non-linear patterns.
- **Ridge and Lasso Regression:** Regularized versions of linear regression that prevent overfitting by penalizing large coefficients.
  - Ridge Regression: Adds an L2-norm penalty term.
  - Lasso Regression: Adds an L1-norm penalty term.

Context: Section 1.2.1 Regression based models, Section 2.1.2 eQTL analysis with MatrixEQTL, Section 2.2 evolution of regression methods for GRN inference**.**

Spearman Correlation:

Definition: The Spearman correlation coefficient ρ measures the strength and direction of the monotonic relationship between two variables. It does not assume that the relationship is linear and can be used with ordinal data.

Formula:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

where

d_i : difference between the ranks of corresponding variables

n : number of observations

Key Characteristics:

- Monotonic Relationship: Measures how well teh relationship between two variables can be descMeasures how well the relationship between two variables can be described using a monotonic function.
- Rank-Based: Uses the ranks of the data rather than the raw data values, making it a non-parametric measure.
- Range: Values range from -1 to 1
  - $\rho = 1$: perfect positive monotonic correlation
  - $\rho = -1$: perfect negative monotonic correlation
  - $\rho = 0$: no monotonic correlation
- Robust to Outliers: Less sensitive to outliers compared to Pearson correlation because it is based on ranks.

Context: Section 1.2.1 similarity measure for correlation networks.

Stochastic Differential Equations (SDEs):

Definition: Stochastic Differential Equations are differential equations that involve both deterministic and stochastic components. They describe the evolution of systems where randomness or uncertainty plays a significant role.

Key Concepts:

- Deterministic Dynamics: Represents the system's deterministic behavior, typically described by ordinary differential equations.
- Stochastic Perturbations: Incorporates random fluctuations or noise, often represented by stochastic processes such as Brownian motion
- Ito's Lemma: A formula used to find the differential of a function of a stochastic process.

Formula:

$$dX_t = a(X_t)\,dt + b(X_t, t)\,dW_t$$

where

X_t : stochastic process.

$a(X_t, t)$: drift term, representing the deterministic component.

$b(X_t, t)$: diffusion term, representing the stochastic component.

Solution Methods:

- Numerical Methods: Stochastic simulation algorithms such as Euler-Maruyama method, Milstein method, and Monte Carlo methods.
- Analytical Methods: Limited to specific cases where closed-form solutions are possible, often involving Ito's calculus.

Context: Section 1.2.1 Differential equation models.

Structural Equation Model (SEM):

Definition: Structural Equation Models are statistical models used to analyze the relationships between observed and latent variables. They combine factor analysis and multiple regression analysis to test complex hypotheses about the relationships between variables.

Key concepts:

- Observed Variables: Measurable variables directly observed or measured in the study.
- Latent Variables: Variables that are not directly observed but inferred from observed variables, representing underlying constructs or factors.
- Paths: Represent the hypothesized relationships between variables, including direct effects and indirect effects.

Components:

- Measurement Model: Describes the relationships between latent variables and their observed indicators.
- Structural Model: Describes the relationships between latent variables themselves and any direct relationships between observed variables.

Estimation Methods:

- Maximum Likelihood Estimation: Most common method used to estimate model parameters, assuming multivariate normality.
- Bayesian Estimation: Utilizes Bayesian inference techniques to estimate parameters, allowing for incorporation of prior knowledge and handling of non-normal data.

Formula:

$$Y_i = BY_{-i} + FX_i + \epsilon_i$$

where

Y_i is the i-th endogenous variables

Y_{-i} represents all the endogenous variables except Y_i

X_i represents the exogenous variables assumed to influence Y_i

B is a matrix of regression coefficients for the endogenous variables Y_{-i}

F is a matrix of regression coefficients for the exogenous variables X_i

ϵ_i is the error term for the i-th equation, defined as a Gaussian vector of mean 0 and variance σ^2 and are independent and identically distributed (i.i.d.)

Context: Section 1.2.4 Methods using gene expression and genetic variants , section 2.1 Structural Equation Model.

Weighted Correlation Network Analysis (WGCNA):

Definition: WGCNA is a systems biology method for describing the correlation patterns among genes across microarray samples. It helps identify modules of highly correlated genes and relate these modules to external sample traits.

Key Steps:

1. Construction of Weighted Network:
   - Correlation Matrix: Calculate the Pearson correlation matrix for all pairs of genes.
   - Adjacency Matrix: Transform the correlation matrix into an adjacency matrix using a soft-thresholding power to emphasize strong correlations while penalizing weak correlations.

     $$a_{ij} = |r_{ij}|^{\beta}$$

     where a_{ij} is the adjacency between genes i and j, r_{ij} is the Pearson correlation coefficient, and β is the soft-thresholding power parameter.
2. Topological Overlap Matrix (TOM): Convert the adjacency matrix into a topological overlap matrix, which measures the overlap in shared neighbors between pairs of genes, providing a more robust measure of interconnectedness.
3. Module Detection: Use hierarchical clustering to group genes into modules based on the TOM. Each module is a cluster of genes with high topological overlap.
4. Module Eigengene Calculation: Summarize each module with the first principal component (module eigengene), representing the module's gene expression profile.
5. Relating Modules to External Traits: Correlate module eigengenes with external sample traits to identify modules significantly associated with these traits.

Applications: Identifying gene modules related to specific biological traits or conditions; studying the interconnectedness and function of gene networks.

Context: Section 1.2.1 Early methods for gene regulatory network inference using gene expression data.