Large-scale inference and imputation for multi-tissue gene expression

Ramon Viñas Torné

Department of Computer Science
University of Cambridge

This thesis is submitted for the degree of Doctor of Philosophy

Churchill College September 2023

Declaration

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface and specified in the text. It is not substantially the same as any that I have submitted, or am concurrently submitting, for a degree or diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. I further state that no substantial part of my dissertation has already been submitted, or is being concurrently submitted, for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. This dissertation does not exceed the prescribed limit of 60 000 words.

Ramon Viñas Torné
September 2023

Abstract

Large-scale inference and imputation for multi-tissue gene expression

Ramon Viñas Torné

Integrating molecular information across tissues and cell types is essential for understanding the coordinated biological mechanisms that drive disease and characterise homoeostasis. Effective multi-tissue omics integration promises a system-wide view of human physiology, with potential to shed light on intra- and multi-tissue molecular phenomena, but faces many complexities arising from the intricacies of biomedical data. This integration problem challenges single-tissue and conventional techniques for omics analysis, often unable to model a variable number of tissues with sufficient statistical strength, necessitating the development of scalable, non-linear, and flexible methods.

This dissertation develops inference and imputation methods for the analysis of gene expression data, an immensely rich and complex biomedical data modality, enabling integration across multiple tissues. The imputation task can strongly influence downstream applications, including performing differential expression analysis, determining co-expression networks, and characterising cross-tissue associations. Inferring tissue-specific gene expression may also play a fundamental role in clinical settings, where gene expression is often profiled in accessible tissues such as whole blood. Due to the fact that gene expression is highly context-specific, imputation methods may facilitate the prediction of gene expression in inaccessible tissues, with applications in diagnosing and monitoring pathophysiological conditions.

The modelling approaches presented throughout the thesis address four important methodological problems. The first work introduces a flexible generative model for the in-silico generation of realistic gene expression data across multiple tissues and conditions, which may reveal tissue- and disease-specific differential expression patterns and may be useful for data augmentation. The second study proposes two deep learning methods to study whether the complete transcriptome of a tissue can be inferred from the expression of a minimal subset of genes, with potential application in the selection of tissue-specific biomarkers and the integration of large-scale biorepositories. The third work presents a novel method, hypergraph factorisation, for the joint imputation of multi-tissue and cell-type gene expression, providing a system-wide view of human physiology. The fourth study proposes a graph representation learning approach that leverages spatial information to improve the reconstruction of tissue architectures from spatial transcriptomic data. Collectively, this thesis develops flexible and powerful computational approaches for the analysis of tissue-specific gene expression data.

Table of contents

Glossary

Biological process Coordinated process that occurs within an organism, cell, or tissue, that is fundamental for the well-functioning of the organism.

Cell Basic building block of life that carries out specialised functions to sustain vital processes, e.g. producing energy and transporting oxygen. Every cell is composed of different organelles.

Cell-type Class of cells with certain morphological or phenotypical features.

DNA Deoxyribonucleic acid. A double-stranded molecule made of 4 types of nucleotides (adenine: A, guanine: G, cytosine: C, thymine: T) that encodes genetic information.

eQTL Expression quantitative trait locus. Variation in the genome that is associated with a particular gene expression trait.

Gene Segment of DNA that may encode instructions to produce proteins and may determine different traits of the organism.

Gene expression Process by which genetic information encoded in the DNA is transformed into functional molecules that carry out vital functions.

Gene regulation Broad range of mechanisms that take place in cells to increase or decrease the production of certain gene products.

Genome Complete set of genetic information in an organism.

Molecular function Event involving molecules that occurs at a molecular level.

mRNA Messenger RNA. RNA molecule that transports genetic information from the DNA into the ribosomes, where proteins are manufactured.

Nucleus Organelle of eukaryotic cells that contains the DNA of the cell.

Omics data Biological information generated from high-throughput techniques such as RNA-sequencing (e.g. transcriptomics). Different types of omics focus on different types of molecules.

Organ Collection of tissues that form a functional unit of the organism that carries out a high-level function, e.g. heart and lungs.

Organelle Structure or compartment of a cell that performs essential tasks like the generation of energy to power biochemical reactions.

Organism Living entity composed of one (unicellular organism) or more cells (multicellular organism).

Phenotype Observable trait or characteristic of an organism.

Protein Large and complex molecules that carry out a broad range of essential functions within cells and organisms.

QTL Quantitative trait locus. Variation in the genome that is associated with a particular phenotype.

RNA Ribonucleic acid. A single-stranded molecule made of 4 types of nucleotides (adenine: A, guanine: G, cytosine: C, uracil: U) that encodes genetic information.

RNA-seq RNA sequencing. Technique that quantifies transcript abundances in single cells and tissues.

Tissue Group of cells that work together to perform a specific function within an organism.

Transcript Copy of a certain fragment of the DNA in the form of RNA.

Transcriptome Complete set of transcripts in a particular biological sample (e.g. cell or tissue).

Transcriptomic data Transcriptomic data measures transcript abundances, allowing the study of gene expression and gene regulation in tissues and single cells.

Chapter 1

Introduction

High-throughput technologies such as RNA sequencing allow us to characterise the biological processes and molecular functions of single cells and tissues in living organisms. This provides a high-resolution picture of molecular states in health and in disease. The resulting omics data — which quantifies different biological molecules in a cell or tissue — is vastly rich and entangled, challenging our ability to discern the patterns underlying the complexities of biology. To address these difficulties, computational and statistical methods can help us make sense of the large amounts of omics data that could otherwise not be processed by the human mind, with potential to unravel the molecular foundations of life.

The analysis of omics data presents numerous challenges for computational approaches which have yet to find successful solutions across datasets and tasks [1]. These challenges revolve around the intricacies of biomedical data (e.g. high dimensionality, redundant features, and noise), experimental settings (e.g. invasive sampling processes and technical confounders), and post-hoc analyses (e.g. interpretability and context-specificity). As such, there is a growing need for robust approaches capable of imputing missing or unreliable values [2]; integrating heterogeneous omics data across modalities [3, 4], tissues [2, 5], experimental settings [6], and species [7]; dealing with high-dimensional data in combination with a scarce number of labelled samples [8]; and interpreting methods to derive novel biological insights [9]. Further methodological efforts may therefore allow us to identify meaningful patterns from omics data, with important applications in drug discovery, medical diagnosis, and precision medicine.

In this dissertation, we introduce computational methods for the analysis of high-throughput transcriptomic data — which measures the expression levels of genes within a cell or tissue — and focus on its tissue-specificity. Understanding gene expression in a context-dependent manner is important because the same genome may generate

uniquely distinct phenotypes in different tissues and cell types [10, 11], allowing them to carry out specialised functions (e.g. production of insulin in the B cells of the pancreas [10]). Thus, characterising biological processes and molecular functions in a context-specific manner might help us elucidate the molecular origins of complex traits with improved resolution.

A central theme of this thesis is the imputation of transcriptomic data: can we infer tissue-specific gene expression as a function of collected molecular information, phenotypes, or demographic covariates? This problem can powerfully influence downstream applications, including performing differential expression analysis, identifying regulatory mechanisms, determining co-expression networks, and enabling drug target discovery [5]. Inferring tissue-specific gene expression may be important in clinical scenarios, where molecular information is often measured in easy-to-acquire tissues such as whole blood (due to their ease of collection), with applications in diagnosing and monitoring pathophysiological conditions. However, gene expression is tissue and cell-type specific [5, 12], limiting the utility of a proxy tissue. Imputation methods may therefore facilitate the prediction of gene expression in difficult-to-acquire tissues, opening the door to a fine-grained characterisation of molecular events.

Throughout the dissertation, we address several challenges of modelling tissue-specific transcriptomic data. We first investigate to what extent we can generate realistic gene expression data in-silico, which may be useful for data augmentation purposes [13] and may shed light on tissue- and disease-specific differential expression [14]. We then study whether the full transcriptome can be reconstructed from a minimal subset of genes, addressing the missing data problem within a single tissue. This is particularly important because missing data can adversely affect downstream analyses [2, 15] and imputation methods might facilitate the integration of large-scale transcriptomic biorepositories [2]. Next, we present a novel methodology for multi-tissue gene expression imputation, enabling the imputation of gene expression in uncollected tissues (e.g. inaccessible tissues such as heart) from a variable number of reference tissues (e.g. accessible tissues like whole blood) of the same individual [5]. In contrast to existing methods, our approach offers a system-wide view of human physiology, incorporating inductive biases to exploit the shared regulatory architecture of tissues and genes. Finally, we build on recent advances in spatial transcriptomic methodologies to analyse the spatial organisation of cells within a tissue, characterising cellular heterogeneity. We propose a spatial deconvolution model that incorporates spatio-relational inductive biases and facilitates an effective spatial reconstruction of

tissue architectures [16]. Altogether, our work offers versatile tools for the analysis of tissue-specific transcriptomic data with a broad range of downstream applications.

1.1 Research questions and contributions

In this thesis, we study the problem of modelling tissue-specific gene expression. We address several challenges that include the in-silico generation of realistic transcriptomic data, the intra- and multi-tissue imputation of gene expression, and the cell-type deconvolution of spatial transcriptomics. In particular, we pose the following research questions:

- **Research question 1:** Can we generate realistic tissue-specific gene expression data *in-silico*?

  Synthetically generated gene expression data is often used for data augmentation and for benchmarking gene expression analysis algorithms, but existing simulators have been criticised because they fail to emulate key properties of gene expression [17]. The problem of generating transcriptomics data is accompanied by the challenging task of assessing its degree of realism — unlike for images, we do not have an intuitive understanding of high-dimensional gene expression.

  In **Chapter 3**, we develop a generative model of transcriptomic data based on Wasserstein generative adversarial networks with gradient penalty [18]. We investigate to what extent the synthetically-generated data preserves key properties of gene expression, including tissue- and cancer-specificity as well as clustering and correlation patterns, and propose novel metrics to evaluate its degree of realism. We also study the application of the proposed method to identify candidate biomarkers for different cancer types.

- **Research question 2:** To what extent can the expression of a subset of genes be used to recover the full transcriptome of a tissue?

  Genes that participate in similar biological processes or that have shared molecular function are likely to have similar expression profiles [19], prompting the question of gene expression prediction from a minimal subset of genes. Gene expression measurements may also suffer from unreliable values because some regions of the genome are extremely challenging to interrogate due to high genomic complexity or sequence homology [20], highlighting the need for accurate imputation.

In **Chapter 4**, we introduce two deep learning methods for gene expression imputation and study their performance on transcriptomic data from a large number of tissues. We compare the proposed methods with existing imputation approaches and evaluate their predictive performance and runtime on the most comprehensive human transcriptome resource available. We further investigate the cross-study generalisation across varying levels of missingness.

- **Research question 3:** Can we impute gene expression of inaccessible tissues as a function of the transcriptome measured at *multiple* accessible tissues?

  Due to the invasiveness of the sampling process, gene expression is usually measured independently in easy-to-acquire tissues such as whole blood [12, 21], leading to an incomplete picture of an individual's physiological state and necessitating effective multi-tissue integration tools. Computational models that exploit multi-tissue patterns could therefore be used to impute the transcriptomes of uncollected tissues (e.g. inaccessible tissues like heart [22]), with potential to elucidate the biological mechanisms regulating a diverse range of developmental and physiological processes.

  In **Chapter 5**, we present a parameter-efficient graph representation learning approach for multi-tissue gene expression imputation. The proposed approach supports a variable number of collected tissues per individual and imposes inductive biases to leverage the shared regulatory architecture of tissues and genes. We study imputation performance using a single reference tissue (whole blood) and multiple reference tissues (accessible tissues). We utilise the fully-imputed dataset to detect regulatory genetic variations (eQTLs) and assess their replicability on independent tissue-specific datasets.

- **Research question 4:** Can we characterise spatial cell-type heterogeneity in tissues using spatial transcriptomic data?

  Analysing the spatial organisation of cells within a tissue can shed light on fundamental biological processes, including intercellular communication [23] and organogenesis [24], and mechanisms of diseases like cancer, diabetes, and autoimmune disorders [25–27]. Computational approaches have been developed to infer fine-grained cell-type compositions across locations, but they frequently treat neighbouring spots independently of each other, raising the question of whether accounting for neighbourhood information can yield improved reconstruction of tissue architectures.

  In **Chapter 6**, we study whether incorporating spatio-relational inductive biases leads to enhanced cell-type mapping in spatial transcriptomic data. We build on the

observation that neighboring spots tend to exhibit similar cell-type compositions to extend a state-of-the-art spatial deconvolution model. We conduct extensive ablation experiments to investigate whether this approach attains improved performance over spatial-agnostic baselines.

Table 1.1 Summary of chapter contents. We use bulk, single-cell, and spatial transcriptomics datasets in different chapters of the dissertation. We propose several methods that can be categorised into single-tissue (i.e. operating on a single tissue sample at a time) vs multi-tissue (i.e. operating on multiple collected tissues of an individual); and generative, self-supervised, deep learning, and graph neural network methods. The proposed approaches enable different downstream applications on tissue-specific gene expression, including simulation of new samples, imputation, deconvolution, and expression Quantitative Trait Loci (eQTL) mapping.

	Data			**Method characteristics**									**Analysis**						
	Bulk transcriptomics	Single-cell transcriptomics	Spatial transcriptomics	Single-tissue	Multi-tissue	Discriminative	Generative	Adversarial	Variational inference	Self-supervised	Multi-layer perceptron	Graph neural nets	Degree of realism	Data integration	Imputation	Deconvolution	Differential expression	Enrichment analysis	eQTL mapping
Chapter 3	•			•			•	•			•		•	•			•	•	
Chapter 4	•			•		•		•		•	•			•	•			•	
Chapter 5	•	•			•	•	•		•	•	•	•		•	•	•	•	•	•
Chapter 6		•	•	•			•		•		•	•		•		•			

We categorise the main topics of each chapter and research question in terms of the transcriptomic data types, developed methods, and downstream analyses in Table 1.1. In addition to the main contributions summarised above, **Chapter 2** covers the background materials, that is, basic notions of gene expression and RNA sequencing, different statistical methods for gene expression analysis used throughout the thesis, probabilistic models of gene expression, and two widely-used unsupervised learning techniques. Finally, **Chapter 7** provides a conclusion, outlining the main developments of the thesis and future directions. Figure 1.1 shows a graphical overview of the thesis.

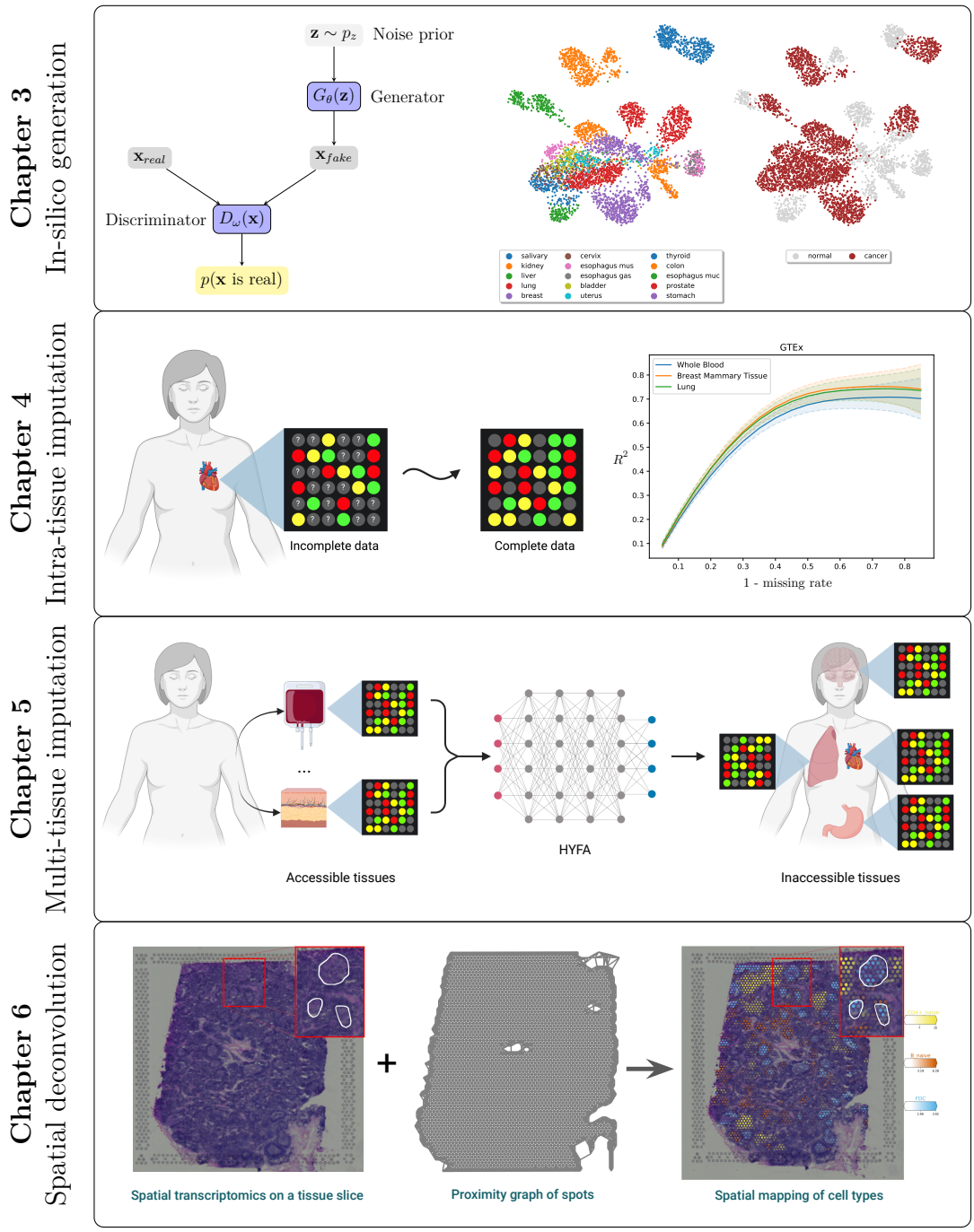


Fig. 1.1 Thesis outline. Chapter 2 introduces the background material, including statistical and probabilistic methods for gene expression analysis. Chapter 3 presents a generative method for simulating gene expression data in-silico. Chapter 4 studies the problem of intra-tissue imputation, wherein the whole-genome gene expression data is reconstructed from a subset of genes. Chapter 5 introduces a method for jointly modelling gene expression collected from a variable number of tissues of a given individual. Chapter 6 investigates the use of spatio-relational inductive biases for cell-type deconvolution in spatial transcriptomics data. Chapter 7 summarises the main contributions of the dissertation.

1.2 Publications

The work presented in this dissertation has been published in the following papers:

- **Ramon Viñas**, Helena Andrés-Terré, Pietro Liò, and Kevin Bryson. Adversarial generation of gene expression data. *Bioinformatics*, 01 2021. `https://doi.org/10.1093/bioinformatics/btab035`
- **Ramon Viñas**, Tiago Azevedo, Eric R. Gamazon, and Pietro Liò. Deep learning enables fast and accurate imputation of gene expression. *Frontiers in Genetics*, 12:489, 2021. `https://doi.org/10.3389/fgene.2021.624128`
- **Ramon Viñas**, Chaitanya Joshi, Dobrik Georgiev, Phillip Lin, Bianca Dumitrascu, Eric R. Gamazon, and Pietro Liò. Hypergraph factorisation for multi-tissue gene expression imputation. *Nature Machine Intelligence*, 2023. `https://doi.org/10.1038/s42256-023-00684-8`
- **Ramon Viñas***, Paul Scherer*, Nikola Simidjievski, Mateja Jamnik, and Pietro Liò. Spatio-relational inductive biases in spatial cell-type deconvolution. *2023 ICML Workshop on Computational Biology*, 2023. `https://doi.org/10.1101/2023.05.19.541474`

The thesis does not cover the following publications authored during the PhD:

- Paris DL Flood, **Ramon Viñas**, and Pietro Liò. Investigating estimated kolmogorov complexity as a means of regularization for link prediction. *2020 NeurIPS workshop on Causality*, 2020.
- Pietro Barbiero*, **Ramon Viñas***, and Pietro Liò. Graph representation forecasting of patient's medical conditions: Toward a digital twin. *Frontiers in genetics*, 12:652907, 2021.
- Viola Fanfani, **Ramon Viñas**, Pietro Liò, and Giovanni Stracquadanio. Discovering cancer driver genes and pathways using stochastic block model graph neural networks. *bioRxiv*, 2021.
- James King, **Ramon Viñas**, Alexander Campbell, and Pietro Liò. An investigation of pre-upsampling generative modelling and generative adversarial networks in audio super resolution. *arXiv preprint arXiv:2109.14994*, 2021.

- Paul Scherer, Maja Trebacz, Nikola Simidjievski, **Ramon Viñas**, Zohrer Shams, Helena Andres Terre, Mateja Jamnik, and Pietro Liò. Unsupervised construction of computational graphs for gene expression data with explicit structural inductive biases. *Bioinformatics*, 38(5):1320–1327, 2022.

- Arian Jamasb, **Ramon Viñas**, Eric Ma, Yuanqi Du, Charles Harris, Kexin Huang, Dominic Hall, Pietro Lió, and Tom Blundell. Graphein - a Python library for geometric deep learning and network analysis on biomolecular structures and interaction networks. In *Advances in Neural Information Processing Systems*, 2022.

- Ben Day*, **Ramon Viñas***, Nikola Simidjievski, and Pietro Liò. Attentional meta-learners for few-shot polythetic classification. In *International Conference on Machine Learning*, pages 4867–4889. PMLR, 2022.

- Han-Bo Li, **Ramon Viñas**, and Pietro Liò. Improving classification and data imputation for single-cell transcriptomics with graph neural networks. In *NeurIPS 2022 AI for Science: Progress and Promises and NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*, 2022.

- Tolou Shadbahr, Michael Roberts, Jan Stanczuk, Julian Gilbey, Philip Teare, Sören Dittmer, Matthew Thorpe, **Ramon Viñas**, Evis Sala, Pietro Liò, et al. Classification of datasets with imputed missing values: Does imputation quality matter? *Nature Communications Medicine*, 2023.

- Dobrik Georgiev*, **Ramon Viñas***, Sam Considine, Bianca Dumitrascu, and Pietro Liò. NARTI: Neural Algorithmic Reasoning for Trajectory Inference. *2023 ICML Workshop on Computational Biology*, 2023.

- Chaitanya K. Joshi, Arian R. Jamasb, **Ramon Viñas**, Charles Harris, Simon Mathis, and Pietro Liò. Multi-state RNA design with geometric multi-graph neural networks. *2023 ICML Workshop on Computational Biology*, 2023.

(shared first co-authorships are marked with an asterisk *)

Chapter 2

Background

2.1 Gene expression

Gene expression analysis is a central theme of this dissertation. In this section, we review the central dogma of molecular biology and the gene expression process. We then present the landscape of omics modalities that enable the study of the central dogma of molecular biology in different layers, including transcriptomics. We also provide an overview of RNA sequencing (RNA-seq), which offers a snapshot of the transcriptome in a biological sample, and discuss some of the main nuisance factors of RNA-seq data.

2.1.1 Central dogma of molecular biology

Gene expression is the process of manufacturing functional molecules, e.g. proteins, from the genetic information encoded in the DNA. These molecules carry out all the functions necessary for life and include, for instance, the enzymes that metabolise nutrients or the DNA polymerases that are responsible for DNA duplication when the cell divides [41]. The flow of genetic information from DNA to functional proteins is often known as the *central dogma of molecular biology* and, for eukaryotic cells, consists of two main steps: transcription and translation.

Transcription The information encoded in the DNA is *transcribed* into a newly assembled fragment of RNA known as messenger RNA (mRNA). During transcription, an enzyme known as RNA polymerase matches the DNA nucleotides of a gene with their complementaries ($A \rightarrow U$; $T \rightarrow A$; $C \leftrightarrow G$), yielding precursor mRNA (pre-mRNA). Then, the introns (noncoding sequences) are removed through a process called

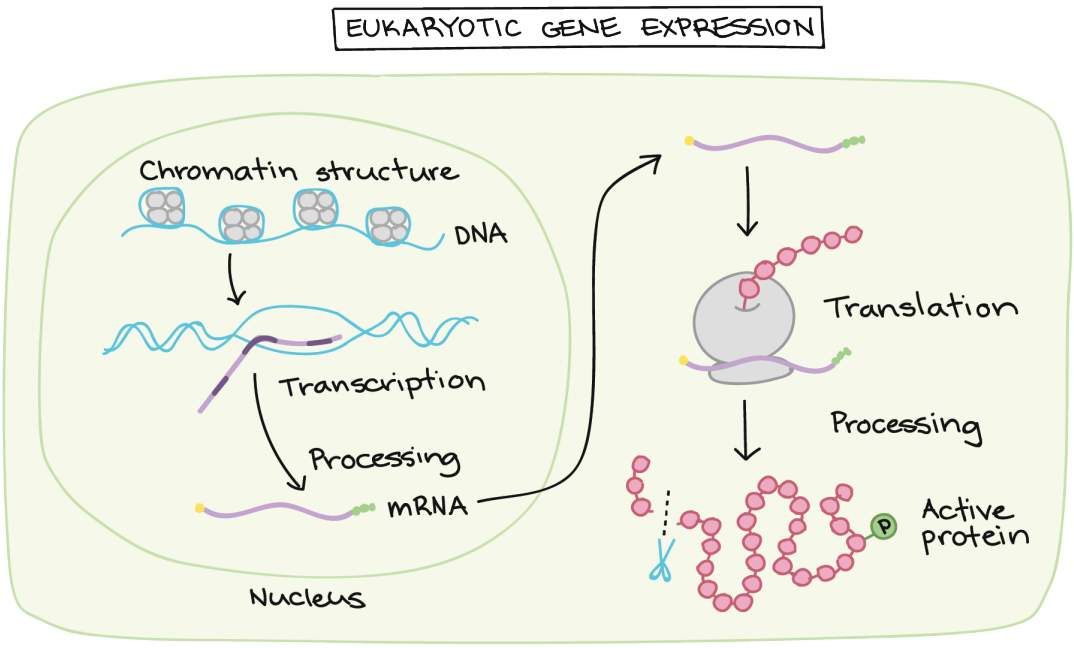


Fig. 2.1 Eukaryotic gene expression. Source: Khan academy [40]

RNA splicing, producing mature mRNA that can be used to synthesise new proteins. Importantly, *alternative splicing* can occur, that is, the same pre-mRNA molecule can be spliced into several types of mRNA fragments that code for different proteins.

Translation After transcription, mature mRNA leaves the nucleus of the cell and travels through the cytoplasm to organelles called ribosomes, where translation takes place. During translation, each codon or triplet of bases in mRNA is matched with the complementary anticodon from transfer RNA (tRNA). These tRNA molecules are physically attached to a specific amino acid according to the genetic code, a dictionary matching each mRNA codon with one of the 20 amino acids. The beginning of the mRNA is known as the untranslated region (UTR) and contains a ribosome-binding site [41] before the start codon (AUG), which triggers the start of the process. As translation progresses, the ribosome assembles the resulting amino acids sequentially into a growing polypeptide chain. This is known as the elongation phase. Once the stop codon is found, the ribosome stops translating and the polypeptide chain is released. Finally, the polypeptide folds into a 3D shape and becomes a functional protein.

2.1.2 Multi-omics

High-throughput technologies provide insights into the inner workings of a cell. They can measure molecular information that underlies cellular states and communication networks in diverse tissues and conditions [42]. These high-resolution *omics* data span multiple molecular layers, each viewing the central dogma of molecular biology from a different perspective:

- **Genomics** reveals the genotypes or DNA sequences of individuals. This allows to identify genetic variants associated with diseases or responses to treatments [43] through what is known as genome-wide association studies (GWAS). Currently, with next generation sequencing (NGS) the whole human genome can be sequenced within a single day [44].

- **Epigenomics** characterises reversible modifications of DNA such as DNA methylation, chromatin accessibility, and histone modifications [43]. Epigenetic modifications play a fundamental role in regulation of gene expression, e.g. by activating or repressing the transcription of genes, and establishing cellular phenotypes [42]. They are often tissue-specific and can sometimes be linked to pathologies such as cancer [45], cardiovascular diseases [43], and neurodegenerative disorders, including Alzheimer's and Parkinson's disease [46]. Epigenomics traits such as chromatin accessibility can be measured with technologies such as single-cell ATAC-seq (scATAC-seq).

- **Transcriptomics** measures the abundance of mRNA, allowing to quantify the activity levels of genes across the entire genome. Currently, RNA abundance can be measured with high resolution through RNA sequencing (RNA-seq). Nowadays, this technology works at single-cell level and allows to understand the heterogeneity of cell types in different tissues [43] as well as the spatial organisation of cells [47].

- **Proteomics** measures protein abundance and how proteins interact with each other, yielding protein-protein interaction networks. It is also possible to detect interactions between proteins and nucleic acids via a technique known as chromatin immunoprecipitation sequencing (ChIP-seq) [43].

- **Metabolomics** quantifies the amount of metabolites or small molecules, e.g. amino acids and carbohydrates, among others [43]. This type of omics data can be used to understand the physiology of the cell because it provides snapshots

about the end products of cellular processes. Metabolomics can be measured via mass spectrometry.

In this dissertation, we mainly focus on transcriptomics, which has generated large-scale databases, including tissue banks [48] and cell atlases [49], and is arguably the most widespread omics modality.

Paired omics modalities In recent years, we have experienced rapid development of experimental technologies for the joint profiling of multiple modalities from the same single cell. For example, we can now use CITE-seq to measure the cell transcriptome and protein levels on the cell surface [50]. Similarly, sci-CAR simultaneously profiles chromatin accessibility and mRNA within single cells [4]. These techniques have potential to uncover biological phenomena that cannot be gleaned from a single modality, including the causal mechanisms of gene regulation. Thus, computational approaches that integrate these modalities may allow us to bridge the gaps between steps of the central dogma of molecular biology.

2.1.3 RNA sequencing

The development of next-generation sequencing (NGS) technologies has brought about accurate readings of nucleotide sequences, including DNA and RNA molecules, in a massively parallel way (i.e. allowing to sequence hundreds to thousands of genes simultaneously). RNA sequencing (RNA-seq) builds on NGS to quantify transcript levels in single cells and tissues, unravelling a broad range of downstream applications such as differential expression analysis, characterisation of co-expression networks, and interpretation of the functional elements of the genome [51].

Measuring transcript levels In general, the RNA sequencing process consists of the following steps:

1. Isolate the RNA molecules from the biological sample of interest, such as tissues (bulk RNA-seq) or cells (single-cell RNA-seq), and break them down into small fragments, usually between 200 and 500 bases long [51]. Single-cell RNA-seq profiles gene expression in a single cell (usually encapsulated into separate droplets [52]), while bulk RNA-seq provides the average gene expression profile of an entire population of cells (e.g. in a tissue).

2. Synthesise complementary DNA (cDNA) molecules — which are more stable than RNA molecules — through a reaction involving reverse transcriptase, an enzyme that transcribes single-stranded RNA into double-stranded DNA.

3. Add sequencing adapters or barcodes for sample identification. This allows the sequencing machine to recognise the fragments and facilitates sequencing different samples at the same time (i.e. different samples use different adapters).

4. Amplify the labelled cDNA fragments, usually through a technique known as polymerase chain reaction.

5. Massively parallel sequencing. The cDNA fragments are simultaneously read by the sequencing machine, yielding millions of reads. There exist multiple high-throughput sequencing platforms, including Illumina [53] and PacBio [54], each with its own advantages and shortcomings.

6. Align the reads to a reference genome and quantify the number of reads mapping to every transcript. This step generates a tabular dataset where each entry denotes transcript abundance (i.e. read counts) in a certain sample.

Short-read vs long-read sequencing Most sequencing methods can be classified into short-read and long-read sequencing. Short-read sequencing methods generate high numbers of short reads, resulting in multiple copies of DNA fragments with low per-base error rates and allowing massive parallelization at low cost. In contrast, long-read sequencing methods produce much longer fragments (typically several kilobases long), allowing to capture complex regions with continuous, uninterrupted reads [55]. In this thesis, we use high-throughput short-read sequencing data.

Correcting for sequencing biases The resulting RNA-seq dataset is susceptible to sequencing biases and nuisance factors, which may hinder downstream applications, necessitating further processing steps. For example, different samples may exhibit differences in the total number of reads, i.e. sequencing depth, and longer genes are expected to have higher read counts [56–58]. Various approaches have been proposed to alleviate these issues, including normalisation via *Reads Per Kilobase per Million* (RPKM) and *Transcripts Per Million* (TPM). RPKM adjusts the number of reads by dividing by the product of the gene length (in kilobases) and the total number of million reads [56, 57]. Transcripts Per Million further normalises RPKMs so that the total counts per sample is a million — this facilitates comparison across samples. More

advanced techniques such as Trimmed Mean of M-values (TMM) clip off the most highly variable genes to calculate a normalisation factor [59] that is more robust to technical factors (e.g. sample contamination).

Batch effects Comparing samples collected under different experimental conditions is problematic because technical sources of variation may act as confounders for true biological differences. This problem is often referred to as *batch effects* and, if left unaddressed, may hinder downstream analyses and lead to invalid conclusions. To mitigate this issue, there exist several computational approaches, including ComBat [60], Mutual Nearest Neighbours [61], and Scanorama [6]. ComBat [60] introduces an empirical Bayes framework to adjust the location and scale of the gene expression data, reducing differences between technical batches. Mutual Nearest Neighbours (MNN) [61] finds mutually similar cells across experimental batches and applies a correction vector based on their expression differences to perform batch effect correction. Scanorama [6] is a similar non-linear technique that successively merges multiple single-cell RNA-seq into a single dataset. Unfortunately, batch correction methods are often susceptible to a plethora of issues, including overcorrection [62] and introduction of spurious group differences [63], and to date there is no definitive solution for this problem.

Bulk, single-cell, and spatial transcriptomics In general, gene expression can be measured from single cells (scRNA-seq) or from an entire population of cells (bulk RNA-seq), i.e. bulk RNA-seq produces a mixture of the transcriptome profiles of the material under study (e.g. a tissue). On the one hand, bulk RNA-seq is suitable for studying high-level relationships and differences between biological entities (e.g. tissues) and conditions (e.g. disease states or treatments). On the other hand, single-cell RNA-seq is useful to investigate the fine-grained biology and cellular heterogeneity of single cells [64]. Spatial transcriptomics is another recently developed technique — named Method of the Year 2020 [65]— that profiles gene expression *in situ*, allowing characterisation of the cellular organisation of tissues, with potential to reveal cellular interactions [66] and identify spatially informative genes [67].

Technical artefacts in single-cell RNA-seq Single-cell RNA-seq is inherently noisy and presents several challenges arising from the sequencing process. Single-cell datasets tend to be notoriously sparse, with the fraction of zeroes being often as high as 90% [68], i.e. for a given cell, many genes do not have any mapped reads [1]. These zeros can be attributed to either true absence of expression (biological zeros) or technical

noise (artificial zeros), leading to a phenomenon often known as *dropout*. This artificial zero-inflation event may occur at several points of the sequencing pipeline and may be caused by mRNA degradation after cell lysis (i.e. when the cell membrane is broken to extract the mRNA), limited efficiency in capturing and converting mRNA molecules into cDNA, or low sequencing depth, among others [69, 70]. In practice, dropouts might hinder downstream analyses on scRNA-seq and generalisation to different sequencing protocols [71]. To alleviate this issue, several statistical approaches have been developed to impute missing values, including MAGIC [72], which denoises the count matrix by sharing information across similar cells, and scImpute [71], which simultaneously identifies and imputes likely dropout events.

Another technical artefact observed in scRNA-seq is referred to as *doublets*, where two cells are wrongly captured within the same droplet. In subsequent analyses, this event can potentially lead to the inaccurate identification of rare cell-types with intermediate transcriptome profiles [69]. To overcome this problem, computational methods such as DoubletFinder [73] and Scrublet [74] have been developed. DoubletFinder predicts doublets from the gene expression features, while Scrublet [74] simulates doublets from the data and utilises a nearest neighbour classifier for detection of droplet events.

2.2 Statistical methods for gene expression analysis

In this section, we review standard statistical methods for gene expression analysis, including differential expression analysis, enrichment analyses, and eQTL discovery. We employ some of these techniques in downstream analyses later in the dissertation.

2.2.1 Differential expression analysis

Differential expression analysis aims to identify genes that exhibit statistically different expression patterns in two or more distinct groups of samples. Using statistical testing, we want to determine whether an observed difference in read counts is statistically significant, i.e. whether it is greater than expected just due to natural random variation [75]. Knowledge about the differentially expressed genes can offer valuable insights into the biological processes underlying the conditions of interest.

To perform differential expression analysis, most established methods [75, 76] employ negative binomial regression (Section 2.3.2) to model the data, followed by a statistical test to evaluate differences in the relative transcript abundances between to conditions. In particular, *edgeR* [76, 77] models the expression x_{ij} of gene j in sample

i as:

$$x_{ij} \sim \mathrm{NB}(x_{i+}\lambda_{kj}, \alpha_j),$$

where NB is the Negative Binomial distribution (Section 2.3.2), x_{i+} is the library size of sample i (i.e. the total sample counts), λ_{kj} are the relative expression values of gene j in the experimental group k to which sample i belongs, and α_j is a dispersion parameter that controls the amount of over-dispersion. The dispersion parameter α_j can be specific to every gene or common across all genes [77], i.e. $\alpha_j = \alpha$ for all j, which may be useful in low sample size settings. The model is optimised via conditional maximum likelihood estimation (Section 2.3.1).

To assess the significance of the differential expression for a gene j across two conditions k_1 and k_2, we test the null hypothesis $H_0 : \lambda_{k_1 j} = \lambda_{k_2 j}$ against the two-sided alternative $H_1 : \lambda_{k_1 j} \neq \lambda_{k_2 j}$ [77, 75] and adjust for multiple-testing.

Overall, in the case of large sample sizes, differential expression could be assessed via non-parametric methods such as permutation tests. However, statistical methodologies such as edgeR [76] or DESeq [75] allow identifying differentially expressed genes in settings where the number of samples per condition (i.e. replicates) is limited [75].

2.2.2 Pathway enrichment analysis

Given a list of genes, e.g. genes that are differentially expressed across two conditions, we may want to understand whether they are related to a certain biological pathway. A pathway is a series of molecular interactions that are related to a certain function, for example, cell signaling. Understanding what pathways are active is important for the biological interpretation of gene expression data — pathway enrichment analysis provides mechanistic insights into the possible underlying biology [78] and allows biologists to formulate hypotheses.

There exist several databases of gene sets that describe the genes involved in a broad range of different biological processes, molecular functions, and cellular components. For example, the Gene Ontology knowledge base [79, 80] is one of the largest sources of information on evidence-supported gene functions [80]. The Kyoto Encyclopedia of Genes and Genomes (KEGG) is another reference database of biological pathways that relate genes to specific high-level functions, including pathways, drugs, and diseases [81, 82].

Over-representation analysis (ORA) is a widely-used technique that addresses the challenge of mapping lists of genes to known biological pathways. To achieve this, ORA

first counts the number of pathway-specific genes present in a given gene list for every biological pathway. It then repeats this process for a background list of genes (e.g. collection of all genes), followed by a statistical test to determine whether the pathway genes are over- or under-represented in the gene list relative to the background. There are several choices for the statistical tests, e.g. tests based on the hypergeometric and binomial distributions. For example, the hypergeometric distribution describes the probability of k successes (i.e. genes belonging to a pathway) in n trials (i.e. number of genes in the gene list) in a population of size N (i.e. total number of background genes) that contains K success states (i.e. number of pathway-related genes in the background list of genes). Through the hypergeometric test, we can calculate over-representation p-values as the probability of k or more successes (i.e. genes belonging to the pathway) in n draws (i.e. number of genes in the gene list). We can then compute False Discovery Rate (FDR) values to account for multiple testing.

ORA techniques are limited in that 1) they potentially require setting a manual threshold on the gene list, 2) they ignore the magnitude and gene ranks of the gene list, 3) they assume that genes are independent of each other, and 4) they assume pathways are independent of each other [78]. Nonetheless, they constitute a simple and useful tool to generate biological insights that is independent of the sequencing methodology (i.e. they only require a list of genes as well as the background) and are therefore widely applicable. Gene Ontology over-representation analysis has been broadly used to characterise the biological functions of groups of genes, including the recent application of identifying the major cell processes of the proteins interacting with SARS-CoV-2 [83].

Studying the biological pathways enriched for different gene sets In Chapter 3, we use Gene Ontology over-representation analysis to relate gene clusters to known biological pathways on data from the RNAseqDB database [84]. We also generate gene expression data *in-silico*, apply clustering to identify gene clusters, and assess whether the same pathways are enriched in gene clusters of the generated data. In Chapter 4, we apply over-representation analysis to uncover the enriched KEGG pathways underlying the best-imputed genes. Similarly, in Chapter 5 we employ over-representation analysis to identify Gene Ontology terms enriched for the best-imputed genes in brain tissues using gene expression from the oesophagogastric junction, which may shed light on the biology of the brain-gut axis.

2.2.3 Gene set enrichment analysis

Gene set enrichment analysis (GSEA) [85] is an approach to interpreting gene expression data that overcomes some of the limitations of over-representation analysis techniques. After differential expression analysis, the resulting list of statistically significant differentially expressed genes might be empty (e.g. if the measurement noise is large relative to the biological variance [85]) or the statistically significant genes may be functionally unrelated. GSEA addresses these issues by considering the entire list of ranked genes, removing the need for a threshold, and studying whether members of a gene set tend to fall in one of the extremes of the ranked list.

Given a ranked gene list and a gene set that we want to test, the GSEA algorithm consists of 3 main steps [85]. First, it calculates a running sum or enrichment score (ES) by descending the sorted list of genes, increasing the score whenever a gene belongs to the gene set and decreasing the score otherwise. The maximum value of the running sum is used as the test statistic. Second, a p-value is calculated through a permutation test, by randomly permuting the class labels (in case of differential expression analysis) or the ranked list of genes. For instance, we can permute the ranked list of genes, calculate the running sum, and calculate a p-value by comparing the maximum ES of our hypothesis versus the ES of these randomly-permuted lists. Finally, when multiple gene sets are studied, we calculate the False Discovery Rate (FDR) to account for multiple testing.

After running the GSEA algorithm, we can get further insights into the important genes by analysing the so-called leading-edge subset, that is, genes from the gene set that occur before the point where the enrichment score is maximum. These are the genes responsible for the enrichment signal [85].

Interpreting model weights in post-hoc analysis In Chapter 5 we use Gene Set Enrichment Analysis to determine the extent to which our model captures known biological pathways. We apply GSEA to the learnt per-gene parameters (ranked by magnitude) and identify a large number of statistically significant enrichments. Interestingly, these analyses show that our multi-tissue gene expression model puts strong emphasis on genes related to signaling pathways, which characterise cell communication, and genes related to transcription factors that control tissue-specific gene expression of many target genes.

2.2.4 eQTL mapping

Gene expression is the intermediate step between the genetic information encoded in our DNA and proteins, which carry out fundamental cellular functions. Expression Quantitative Trait Loci (eQTL) studies aim to elucidate genomic variations, e.g. single-nucleotide polymorphisms (SNPs), that are significantly associated with gene expression [86]. Among other applications, eQTL analysis can reveal variants affecting gene regulation and their influence on complex human diseases [87].

eQTL mapping is an approach to identifying genomic variants associated with gene expression (eQTLs). In eQTL mapping studies, the genetic factors associated with gene expression can be classified into proximal or *cis*-eQTLs, eQTLs in the vicinity of the target genes, and distal or *trans*-eQTLs, eQTLs found in distant regions of the genome [87]. There are several approaches for detecting eQTLs, including methods that perform separate tests for every transcript-SNP pair [86] and methods that attempt to identify groups of SNPs [88]. In the most simple form, detecting eQTLs involves fitting a linear regression model for every gene-SNP pair:

$$y = \alpha + \beta s + \boldsymbol{\gamma c} + \epsilon \qquad \epsilon \sim \mathcal{N}(0, \sigma^2),$$

where y is the gene expression of the target gene; s is the encoded SNP; $\boldsymbol{c}$ are covariates that allow accounting for clinical variables (e.g. age and sex); α, β and $\boldsymbol{\gamma}$ are learnable parameters; and ϵ is some random additive noise. The SNP s is encoded as 0, 1, or 2 according to the frequency of the minor allele [86]. Alleles are the possible SNP variations in a particular position of the genome and can be classified into 1) homozygous (two copies of the same allele) vs heterozygous (two different alleles) and 2) major (most common allele in the population) vs minor (less common allele in the population).

After fitting the linear model, we can calculate a statistic (e.g. t-statistic) to test the null hypothesis that the slope β is equal to 0 (i.e. no association between the SNP and the gene expression of the target gene), followed by the calculation of p-values and correction for multiple hypothesis testing (e.g. by calculating the False Discovery Rate). A SNP is then said to be an eQTL for a particular target gene if we are able to reject the null hypothesis.

Discovering new tissue-specific eQTLs In Chapter 5, we apply eQTL mapping to uncover a large number of previously undetected tissue-specific eQTLs.

2.3 Probabilistic modelling of gene expression

In this section, we introduce concepts related to probabilistic models, which are paramount for modelling gene expression. We also examine the probability distributions frequently used to capture the characteristics of transcriptomic data.

2.3.1 Probabilistic models

Probabilistic models allow us to express our beliefs and uncertainties about different phenomena. They are characterised by probability distributions that describe the relationship between different random variables, e.g. how likely is it that it will rain today given that the atmospheric pressure is low? Probability distributions can be employed to make predictions of a certain event happening given a series of observations (supervised scenario) and infer hidden variables governing the observed data (unsupervised scenario), among others.

Supervised scenario In the supervised setting, we assume we are given some observations $\mathbf{x}$ and we want to infer a response variable $\mathbf{y}$. We can model the relationship between the two variables using a conditional model $p_\theta(\mathbf{y}|\mathbf{x})$ with parameters θ. This conditional distribution assigns a probability — or density, in case of a continuous outcome — to every possible value of $\mathbf{y}$ given the observations $\mathbf{x}$, allowing us to estimate the most likely response as well as the probability of alternative outcomes. Given a dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ with n observations, conditional models are usually optimised by maximising the conditional likelihood:

$$\theta^* = \arg\max_\theta \prod_{i=1}^n p_\theta(\boldsymbol{y}_i|\boldsymbol{x}_i) = \arg\max_\theta \sum_{i=1}^n \log p_\theta(\boldsymbol{y}_i|\boldsymbol{x}_i)$$

In other words, the goal is to find the parameters of the model that maximise the conditional likelihood of our data. Importantly, this encodes the assumption that samples are independent and identically distributed, that is, all samples follow the same probability distribution and are mutually independent.

Unsupervised scenario In the unsupervised setting, we assume that the given observations $\mathbf{x}$ depend on some latent, unobserved variables $\mathbf{z}$. The joint probability distribution of the obervations and latent variables can be written as:

$$p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{x}|\mathbf{z})p_\theta(\mathbf{z}),$$

where $p_\theta(\mathbf{x}|\mathbf{z})$ is the likelihood of the observations given the latent variables and $p_\theta(\mathbf{z})$ is known as the *prior* and captures our prior belief on the marginal distribution of the latent variables. This model also belongs to the category of generative models — given the learnt parameters θ we can generate new observations from the joint distribution by first sampling from the prior $p_\theta(\mathbf{z})$ and then from the conditional distribution $p_\theta(\mathbf{x}|\mathbf{z})$. To learn the parameters θ of the model, we can maximise the marginal likelihood $p_\theta(\mathbf{x})$ of the observed data:

$$\theta^* = \arg\max_\theta \prod_{i=1}^{n} p_\theta(\boldsymbol{x}_i) = \arg\max_\theta \prod_{i=1}^{n} \int p_\theta(\boldsymbol{x}_i, \boldsymbol{z}) d\boldsymbol{z},$$

For a given observation $\boldsymbol{x}_i$, we may also infer the posterior distribution over the latent variables:

$$p_\theta(\boldsymbol{z}_i|\boldsymbol{x}_i) = \frac{p_\theta(\boldsymbol{x}_i, \boldsymbol{z}_i)}{\int p_\theta(\boldsymbol{x}_i, \boldsymbol{z}) d\boldsymbol{z}}$$

Unfortunately, the integral $\int p_\theta(\boldsymbol{x}_i, \boldsymbol{z}) d\boldsymbol{z}$ in these equations is often computationally intractable because the integration is performed over all possible values of the multivariate latent variables. Some notable exceptions include Factor Analysis and Probabilistic Principal Component Analysis [89], where the marginal likelihood can be calculated in closed-form. In cases where integrating is unfeasible, various algorithms, such as variational inference (Section 2.4.1), can be used to approximate the marginal likelihood.

2.3.2 Probability distributions for gene expression data

To model gene expression data, we consider several distributions that might be suitable depending on the type of data (e.g. bulk or single-cell RNA-seq) and the processing techniques applied (e.g. inverse-normal transformed vs raw read counts).

Normal distribution The Normal (or Gaussian) distribution is a fundamental probability distribution for modelling continuous data. This distribution, characterised by a symmetric bell-shaped curve, is ubiquitous in nature — it is commonly used to model many real-world phenomena including biological traits and measurement errors. It is particularly important because of the *central limit theorem*, which states that the average of a large number of independent and identically distributed random variables tends to follow a Normal distribution.

The Normal distribution $\mathcal{N}(\mu, \sigma^2)$ is parameterised by a mean parameter $\mu \in \mathbb{R}$, the expected value, and a dispersion parameter $\sigma^2 \in \mathbb{R}_{>0}$, the variance of the distribution. The probability density function (PDF) of the Normal distribution is given by:

$$f_{\text{Normal}}(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

The Normal PDF is often used as a likelihood function for regression tasks. It is tightly connected to the mean squared error $\text{MSE}(\theta) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - g_\theta(\boldsymbol{x}_i)\right)^2$ commonly used to optimise a broad range of regression models g_θ (e.g. non-linear neural networks). In particular, if we assume a probabilistic model $p_\theta(\text{y}|\mathbf{x}) = \mathcal{N}(\mu = g_\theta(\mathbf{x}), \sigma^2)$ with a Normal likelihood and fixed variance σ^2, we can see that maximising the likelihood of the data $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ under the conditional model $p_\theta(\text{y}|\mathbf{x})$ is equivalent to minimising the MSE:

$$\arg\max_\theta \prod_{i=1}^{n} p_\theta(y_i|\boldsymbol{x}_i) = \arg\max_\theta \sum_{i=1}^{n} \log p_\theta(y_i|\boldsymbol{x}_i) = \arg\min_\theta \sum_{i}^{n} \left(y_i - g_\theta(\boldsymbol{x}_i)\right)^2,$$

where the last equality follows from the definition of the Normal PDF.

The Normal distribution has been widely used to model gene expression data, particularly for quantifying differential expression patterns [90, 91], inferring cell-type composition in bulk gene expression [92, 93], and modelling latent sources of variation [94, 95].

Gamma distribution The Gamma distribution is a continuous probability distribution that, intuitively, models the wait time until the k-th event occurs for a given rate of occurrence. It is parameterised by a shape parameter $k \in \mathbb{R}_{>0}$, controlling the spread of the distribution (i.e. the more events, the longer the wait time), and a scale parameter $\theta \in \mathbb{R}_{>0}$ (inverse of the occurrence rate). The probability density function of the Gamma distribution is:

$$f_{\text{Gamma}}(x; k, \theta) = \frac{x^{k-1}e^{-x/\theta}}{\theta^k \Gamma(k)},$$

where $\Gamma(k) = \int_0^\infty t^{k-1}e^{-t}dt$ is the gamma function and corresponds to the factorial $\Gamma(n) = (n-1)!$ for all positive integers $n \in \mathbb{N}_{>0}$.

When $k = 1$, the Gamma distribution reduces to the exponential distribution, which models the distribution of time between two events occurring at a constant average rate. This distribution has broad applications in Bayesian statistics and serves

as the conjugate prior for many probability distributions including Normal, Poisson, and exponential distributions. For example, in Cell2location [96], a framework to infer the cell-type composition of spatial transcriptomic spots, the Gamma distribution is used as a prior for the per-spot cell-type abundances.

Poisson distribution The Poisson distribution models the probability of an event happening a certain number of times k in a fixed interval of time or space. It is a discrete probability distribution that has been broadly used to model the occurrence of rare events, including the number of RNA molecules observed for a certain gene in a pool of transcripts.

The Poisson distribution is parameterised by $\lambda \in \mathbb{R}_{\geq 0}$, the average rate of occurrences within a fixed interval, which also corresponds to the mean and variance of the distribution. The probability mass function (PMF) of the Poisson distribution is:

$$f_{\text{Poisson}}(x; \lambda) = \frac{\lambda^x e^{-\lambda}}{x!}$$

In the bioinformatics literature, the Poisson distribution has been used to capture the per-gene variation across technical replicates [97], identify differentially expressed genes [97, 76, 98], and cluster of RNA-seq data [99, 100], among others.

A limitation of the standard Poisson distribution is that it assumes that the mean and variance are the same, rendering the distribution inappropriate in the case of over- (or under-) dispersion. This is often the case for RNA-seq data, e.g. when the variance of a particular gene is larger than its mean. To address this challenge, the negative Binomial distribution allows adjusting the variance independently of the mean.

Negative binomial distribution The negative binomial (NB) distribution generalises the Poisson distribution by introducing an additional parameter with increased flexibility to model over-dispersed data. Intuitively, the negative binomial distribution models the number of independent Bernoulli trials, each with a probability of success $p \in [0, 1]$, needed until reaching a fixed number of $r \in \mathbb{N}_{>0}$ successes. The PMF of the negative binomial distribution is:

$$f_{\text{NB}}(x; r, p) = \binom{x + r - 1}{x} (1 - p)^x p^r,$$

where $\binom{x+r-1}{k} = \frac{(x+r-1)!}{(r-1)!x!}$ is the binomial coefficient representing the number of ways in which k failures can be chosen from a total of $x + r - 1$ trials (the last trial is always a

success). The second and third factors $(1-p)^k p^r$ capture the probability of observing k failures and r successes in any given order. The mean of the distribution is $\frac{r(1-p)}{p}$ and the variance is $\frac{r(1-p)}{p^2}$.

By using the gamma function $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$, which corresponds to the factorials $\Gamma(n) = (n-1)!$ for all positive integers $n \in \mathbb{N}_{>0}$, we can extend the negative binomial PMF to positive real-valued $r \in \mathbb{R}_{>0}$ parameters:

$$f_{\mathrm{NB}}(x; r, p) = \frac{\Gamma(x+r)}{x!\Gamma(r)}(1-p)^x p^r$$

This is particularly useful for regression models, e.g. gradient-based methods, that attempt to approximate the distribution's parameters from the observed data. In particular, it is common to reparameterise the negative binomial PMF in terms of the mean μ and dispersion (or shape) α parameters [101]:

$$f_{\mathrm{NB}}(x; \mu, \alpha) = \frac{\Gamma(x+\alpha^{-1})}{x!\Gamma(\alpha^{-1})}\left(\frac{\mu}{\alpha^{-1}+\mu}\right)^x\left(\frac{\alpha^{-1}}{\alpha^{-1}+\mu}\right)^{\alpha^{-1}},$$

where $\mu = \frac{r(1-p)}{p}$ and $\alpha = \frac{1}{r}$. The mean of the distribution now corresponds to μ and the variance is $\mu + \alpha\mu^2$. The parameter α therefore controls the over-dispersion levels and the Poisson distribution arises as a special case of negative binomial when $\alpha \to 0$ (i.e. no over-dispersion).

The negative binomial distribution can alternatively be viewed as a mixture of Poisson distributions with different means, also known as the Gamma-Poisson distribution. In this case, we use a Poisson distribution $\mathrm{Poisson}(\lambda)$ where the rate parameter λ is a random variable that follows a Gamma distribution $\Gamma(k, \theta)$ with shape parameter $k = \alpha^{-1}$ and scale $\theta = \frac{p}{1-p}$. This is intuitively appealing for modelling RNA-seq counts because the transcripts of different genes may occur at different rates.

The negative binomial distribution is most commonly used to model *bulk* gene expression datasets because it is flexible enough to account for over-dispersed genes. It has been employed for differential expression analysis [76, 102], feature selection [103], and gene expression normalisation [59, 58, 104], among others.

Zero-inflated negative binomial Single-cell RNA-seq data is characterised by its sparsity, i.e. the fraction of zeroes is often as high as 90% and many genes do not have any mapped reads [68, 1]. The abundance of zeros can be explained by several factors, including technical factors such as limited capture efficiency and low sequencing depth. This hinders our ability to distinguish between actual biological zeros and

technical artefacts. For single-cell RNA-seq data, the negative binomial distribution is not flexible enough to model the excess of zeros.

To alleviate this problem, we can model the data as a mixture of two distributions — the first distribution produces zeros (i.e. zero-inflation) and the second produces the actual counts (i.e. via negative binomial distribution). The zero-inflated negative binomial (ZINB) achieves this by introducing an additional parameter $\pi \in [0, 1]$ that captures the probability of inflation or probability of a technical zero (also known as dropout probability). The zero-inflated negative binomial PMF is:

$$f_{\text{ZINB}}(x; r, p, \pi) = \begin{cases} \pi + (1-\pi) f_{\text{NB}}(x; r, p), & \text{if } x = 0 \\ (1-\pi) f_{\text{NB}}(x; r, p), & \text{otherwise} \end{cases}$$

This PMF accounts for the chance of a zero being a technical zero — if $x = 0$, then it's a technical zero with probability π and biological zero with probability $1 - \pi$. The mean of the distribution is now $(1-\pi)\frac{r(1-p)}{p}$ and the variance is $(1-\pi)\frac{r(1-p)(1+r\pi-r\pi p)}{p^2}$. The zero-inflated negative binomial distribution is therefore an excellent choice for modelling single-cell RNA-seq and is used as the preferred likelihood function in the latest single-cell RNA-seq analysis methods, including single-cell variational inference (scVI) [94], deep count autoencoders (DCA) [105], and zero-inflated negative binomial-based wanted variation extraction (ZINB-WaVE) [106].

Separating measurement and expression models Observed RNA-seq counts reflect both true gene expression (biological variation) and measurement error (technical variation). Distinguishing between these two sources of variation may improve clarity on the underlying method assumptions [107]. Sarkar A. and Stephens M. [107] propose a clear separation between (1) an expression model, which describes the variation of true expression counts, and (2) a measurement model, which describes the discrepancy between the observed and true RNA-seq counts. For example, the observed expression levels x_{ij} of gene j in cell i may be modelled as $x_{ij} \sim \text{Poisson}(x_{i+}\lambda_{ij})$, where x_{i+} are the total cell counts (or library size) and λ_{ij} is the true expression modelled as $\lambda_{ij} \sim g_j(\cdot)$. The distribution of the expression model g_j can be chosen based on our assumptions — e.g., using a Gamma distribution for g_j leads to a negative binomial observation model.

2.4 Unsupervised learning with generative models

Generative models are a class of statistical models that allow to discover latent, unobserved variables that drive the data generation process. In this section we describe two families of generative models, variational autoencoders and generative adversarial networks, as well as extensions of these families that are relevant to this dissertation.

2.4.1 Variational inference and variational autoencoders

Variational inference Variational inference is a technique that allows approximating the posterior distribution over the latent variables describing the data, which is often computationally intractable. Formally, let $\mathbf{x}$ and $\mathbf{z}$ be two random variables representing the observed and latent variables, respectively. Let p_θ and q_ϕ be two probability density functions with parameters θ and ϕ. The ELBO loss $\mathcal{L}_{ELBO}$ is defined as:

$$\mathcal{L}_{ELBO} = \mathbb{E}_{q_\phi(\mathbf{z})}[\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q_\phi(\mathbf{z})] = \mathbb{E}_{q_\phi(\mathbf{z})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}\Big(q_\phi(\mathbf{z})||p_\theta(\mathbf{z})\Big),$$

where KL is the Kullback-Leibler divergence or relative entropy.

The ELBO loss can be derived by introducing a variational distribution $q_\phi(\mathbf{z})$ and a lower bound on the log-likelihood $p_\theta(\mathbf{x})$ based on Jensen's inequality (Supplementary Information A). We can approximate the ELBO and gradients with respect to the parameters θ and ϕ via Monte Carlo estimates (i.e., by drawing several random samples from $q_\phi(\mathbf{z})$). Optimising θ and ϕ via stochastic gradient descent on the ELBO is often known as stochastic variational inference.

Variational autoencoders Variational autoencoders (VAEs) are a class of *amortised* variational inference methods for learning deep latent representations [108, 109]. The term *amortised* refers to the fact that the same set of parameters is used to approximate the posterior for all data points. They consist of two coupled models that support each other [109]. One model, the *encoder* or *recognition* model $q_\phi(\mathbf{z}|\mathbf{x})$, approximates the posterior over the latent variables given the observed variables. The other model, the *decoder* or *generative* model $p_\theta(\mathbf{x}|\mathbf{z})$, estimates the conditional probability of the observed variables given the latent variables. Similar to standard variational inference, VAEs work by maximising the evidence lower bound (ELBO):

$$\mathcal{L}_{ELBO} = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}\Big(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})\Big)$$

The main difference is that the variational distribution $q_\phi(\mathbf{z}|\mathbf{x})$ is now conditioned on the observed data, allowing us to learn a mapping between data points and latent variables. Intuitively, the first term measures the reconstruction error, whereas the second term is a regulariser that encourages the variational distribution q to be close to a predetermined prior distribution over the latent variables (e.g. typically an isotropic normal distribution). To balance the encoder's capacity versus the degree of disentanglement, β-VAEs [110] introduce an hyperparameter β that weighs the regularisation strength (i.e. second term of the ELBO).

When we optimise the ELBO loss, backpropagation is not possible by default because the gradients cannot flow through the sampling operation involved in computing the expectation with respect to the variational distribution $q_\phi(\mathbf{z}|\mathbf{x})$. To overcome this issue, VAEs employ the *reparameterisation trick* [108, 109], which consists of externalising the randomness of the sampling $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ by reparameterising the latent variable as a deterministic and differentiable function of ϕ. For example, suppose that $q_\phi(\mathbf{z}|\mathbf{x})$ takes the form of a normal distribution $\mathcal{N}(\boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma}^2))$, where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ depend on ϕ. Then, we can sample a new variable $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and compute $\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\epsilon} \odot \boldsymbol{\sigma}$, rendering the sampling operation differentiable with respect to ϕ.

2.4.2 Generative adversarial networks

Generative Adversarial Networks (GANs) are a framework for estimating generative models via an adversarial process [111]. They are often described as a two-player game in which both players are encouraged to improve. One player, the *generator*, creates samples that are intended to be indistinguishable from those coming from a certain target data distribution. The other player, the *critic*, learns to determine whether samples come from the *adversarial* distribution (*adversarial* samples) or the data distribution (*real* samples). Figure 2.2 shows the basic idea of GANs.

These two players are represented by $D_\omega(\mathbf{x})$ and $G_\theta(\mathbf{z})$, where $\mathbf{z}$ is randomly sampled from a fixed noise distribution p_z (e.g. an isotropic Gaussian with unit variance) and $D_\omega(\mathbf{x})$ indicates the probability of $\mathbf{x}$ coming from the data distribution, e.g. $\mathbf{x} \sim p_r$, as opposed to being generated by the generator, e.g. $\mathbf{x} = G_\theta(\mathbf{z})$. These functions are differentiable with respect to their parameters ω and θ, and in the GAN framework they are represented by neural networks. The model is optimised via the following minimax game [111]:

$$\min_\theta \max_\omega \mathop{\mathbb{E}}_{\mathbf{x}\sim p_r} [\log D_\omega(\mathbf{x})] + \mathop{\mathbb{E}}_{\mathbf{z}\sim p_z} [1 - \log D_\omega(G_\theta(\mathbf{z}))]$$

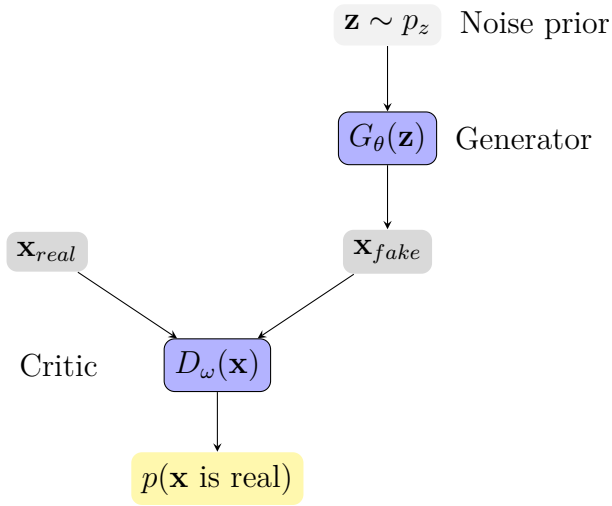


Fig. 2.2 Generative Adversarial Network framework. The generator $G_\theta(\mathbf{z})$ receives a vector $\mathbf{z}$ sampled from a noise prior distribution $p_\mathbf{z}$, and generates a synthetic sample $\mathbf{x}_{fake}$. The critic $D_\omega(\mathbf{x})$ (also known as discriminator) tries to distinguish *real* samples from *fake* samples, producing the probability of $\mathbf{x}$ coming from the *real* data distribution. The competition between the two players drives the game and makes both players increasingly better.

The minimax game can also be described via two loss functions $J_D(\omega, \theta)$ and $J_G(\omega, \theta)$ that are minimised adversarially with respect to ω and θ, respectively:

$$J_D(\omega, \theta) = -\mathop{\mathbb{E}}_{\mathbf{x}\sim p_r}[\log D_\omega(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{z}\sim p_z}[1 - \log D_\omega(G_\theta(\mathbf{z}))]$$
$$J_G(\omega, \theta) = \mathop{\mathbb{E}}_{\mathbf{z}\sim p_z}[1 - \log D_\omega(G_\theta(\mathbf{z}))]$$

Intuitively, the game combines the cross-entropy losses for both the *real* and the *adversarial* data. In other words, the first term of $J_D(\omega, \theta)$ penalises D_ω for labelling *real* data as *synthetic*, while the second term of $J_D(\omega, \theta)$ penalises D_ω for classifying *synthetic* data as *real*. The solution to this game (ω, θ) is a local minima corresponding to a Nash equilibrium [112].

Although this approach is theoretically sound, in practice it has some problems with gradient-based methods, because when the critic successfully rejects *adversarial* samples the generator's cost function $J_G(\omega, \theta)$ saturates and its gradients become too weak [111]. For this reason, it is more common to define the generator's cost as:

$$J_G(\omega, \theta) = -\mathop{\mathbb{E}}_{\mathbf{z}\sim p_z}[\log D_\omega(G_\theta(\mathbf{z}))] \tag{2.1}$$

For this cost function, the generator's gradients are strong when the critic is not fooled by the generator's samples.

Conditional GANs Conditional GANs are a simple extension of the GAN framework to approximate conditional distributions [113]. Conditional GANs include the covariates $\mathbf{y}$ that we wish to condition on as input to both the critic and generator. The minimax game is then defined as:

$$\min_{\theta} \max_{\omega} \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{y})\sim p_r} \Big[\log D_\omega(\mathbf{x},\mathbf{y}) + \mathop{\mathbb{E}}_{\mathbf{z}\sim p_z} [1 - \log D_\omega(G_\theta(\mathbf{z},\mathbf{y}),\mathbf{y})] \Big]$$

We can then fix the covariates $\mathbf{y}$ and sample from $G_\theta(\mathbf{z},\mathbf{y})$ to obtain synthetic samples from a desired class $\mathbf{y}$.

Wasserstein GANs One limitation of traditional GANs is that they are really hard to train. Concretely, [114] showed that when we use the generator's cost from Equation 2.1, the norm of the generator's gradient rapidly increases as the critic gets closer to optimality, resulting in unstable gradient updates. Another widely known problem of GANs is mode collapse, wherein the generator learns to produce samples from a small set of modes that seem plausible to the critic.

Wasserstein GANs (WGANs) address these issues by introducing a cost function based on the *Earth Mover's* distance [115], making the gradients smoother everywhere and allowing us to train the critic until optimality at each training iteration (as opposed to balancing the generator and critic's capacity). This improves the stability of training and has been seen to reduce mode collapse drastically [115]. In contrast to traditional GANs, the output of the critic [115] is unbounded.

Formally, WGANs optimise the following minimax game based on the *Earth Mover's* distance and the Kantorovich-Rubinstein duality [116]:

$$\begin{aligned} &\min_{\theta} \max_{\omega} \mathop{\mathbb{E}}_{\mathbf{x}\sim p_r} [D_\omega(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{z}\sim p_\mathbf{z}} [D_\omega(G_\theta(\mathbf{z}))] \\ &\text{subject to} \qquad ||D_\omega(\mathbf{x}_i) - D_\omega(\mathbf{x}_j)|| \leq ||\mathbf{x}_i - \mathbf{x}_j|| \qquad \forall \mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^n, \end{aligned} \tag{2.2}$$

where the constraint enforces the critic D_ω to be 1-Lipschitz, that is, the norm of the critic's gradient with respect to $\mathbf{x}$ must be at most 1 everywhere. To enforce this constraint, WGAN clips the weights of the critic, forcing them to lie within a predefined range (e.g. $[-0.01, 0.01]$).

Wasserstein GANs with gradient penalty The solution of using weight clipping to enforce the 1-Lipschitz constraint is not ideal. When the clipping hyperparameter is too large, it may become hard to optimise the critic until optimality [115]. Conversely, when the hyperparameter is too small, this solution might lead to vanishing gradients [115]. This often prevents the model from converging and, as a result, the generated samples have poor quality [18].

To alleviate this issue, [18] introduce a way to enforce the 1-Lipschitz constrain by penalising the norm of the critic's gradient, giving raise to WGANs with gradient penalty (WGAN-GPs). Formally, WGAN-GPs solve the minimax problem described in Equation 2.2 as follows:

$$\min_{\theta} \max_{\omega} \mathop{\mathbb{E}}_{\mathbf{x}\sim p_r} [D_\omega(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{z}\sim p_\mathbf{z}} [D_\omega(G_\theta(\mathbf{z}))] - \lambda \mathop{\mathbb{E}}_{\tilde{\mathbf{x}}\sim p_{\tilde{\mathbf{x}}}} \Big(||\nabla_{\tilde{\mathbf{x}}} D_\omega(\tilde{\mathbf{x}})||_2 - 1 \Big)^2,$$

where λ is a user-definable hyperparameter and the samples $\tilde{\mathbf{x}}$ from the distribution $p_{\tilde{\mathbf{x}}}$ are random points along straight lines that connect pairs of real and adversarial samples, that is, $\tilde{\mathbf{x}} = \alpha\mathbf{x} + (1-\alpha)\hat{\mathbf{x}}$ where $\mathbf{x}$ is real, $\hat{\mathbf{x}}$ is synthetic, and $\alpha \sim \mathbb{U}(0,1)$. Intuitively, since enforcing the 1-Lipschitz constraint everywhere is intractable (Equation 2.2), the gradient penalty term is a relaxed version of the constraint that experimentally results in good performance [18].

Chapter 3

In-silico generation of tissue-specific gene expression

Over the last two and a half decades, the emergence of technologies such as spotted microarrays [117], Affymetrix microarrays [118], and RNA-seq [119] has enabled the expression level of thousands of genes from a biological sample to be simultaneously measured, but datasets of an appropriate size are often unavailable. In these cases, synthetically generated data is often used to benchmark gene expression analysis algorithms. An important example of this is evaluating algorithms that reverse engineer gene regulatory networks (GRNs) from transcriptomics data [120–122]. Benchmarking the performance of these methods is challenging because we often lack well-understood biological networks to use as gold standards. As a result, the current approach is to generate synthetic transcriptomic datasets from well-characterised networks [123, 124]. However, current simulators have been criticised because they fail to emulate key properties of gene expression data [17], suggesting that GRN reconstruction algorithms that perform well on synthetic datasets might not necessarily generalise well on real data.

In this chapter, we study the problem of generating realistic transcriptomics data *in-silico*. This is a challenging task because biological systems are highly complex and it is not clear how biological elements interact with each other. Moreover, it is difficult to determine to what extent the expression data generated by a simulator is realistic — unlike in other domains such as image generation, wherein one can qualitatively assess whether an image is realistic, we do not have an intuitive understanding of

The research presented in this chapter has been conducted in collaboration with Helena A. Terré, Kevin Bryson, and Pietro Liò

high-dimensional expression data. To generate gene expression *in-silico*, we develop a model based on a Wasserstein generative adversarial network with gradient penalty (WGAN-GP; [18]). In contrast to existing gene expression simulators such as SynTReN [123] or GeneNetWeaver (GNW; [124]), our model learns to approximate the expression manifold in a data-driven way and does not require the underlying GRN as input. Furthermore, our approach integrates sample covariates such as age, sex, and tissue type (global determinants of gene expression; [125]) to account for their non-linear effects.

As a first case study, we investigate to what extent the proposed framework preserves statistical properties of GRNs. To that end, we develop a transcriptomics simulator for the *E. coli* bacterium, which has the largest amount of experimentally validated regulatory interactions of any organism [126]. We show that our model conserves several gene expression properties significantly better than widely used simulators such as SynTReN or GeneNetWeaver. In particular, we introduce several correlation-based metrics to assess the quality of the synthetic data and find that SynTReN and GeneNetWeaver poorly preserve correlations between transcription factors and target genes. This is undesirable and has important implications on the assessment of the ability of GRN reconstruction algorithms to generalise to real data.

As a second case study, we examine whether our approach can be used to generate realistic human gene expression data. We train our model on human RNA-seq data from the Genotype-Tissue Expression (GTEx) and The Cancer Genome Atlas (TCGA) and produce data that preserves the tissue and cancer-specific properties of transcriptomics data. Moreover, we observe that the synthetic data conserves gene clusters and ontologies both at local and global scales, suggesting that the model learns to approximate the gene expression manifold in a biologically meaningful way. Finally, we propose a tool that leverages the *in-silico* simulator to find *candidate* causal biomarkers for a variety of cancer types.

3.1 Methodology

In this section, we introduce our approach to generating realistic gene expression data. We use script letters to denote sets (e.g. $\mathcal{D}$), upper-case bold symbols to denote matrices or random variables (e.g. $\mathbf{X}$) and lower-case bold symbols to denote column vectors (e.g. $\mathbf{x}$ or $\bar{\mathbf{q}}_j$). The rest of the symbols (e.g. $\bar{q}_{jk}$, G, or f) denote scalar values or functions.

3.1.1 Problem formulation

Consider a dataset $\mathcal{D} = \{(\mathbf{x}, \mathbf{r}, \mathbf{q})\}$ of samples from an unknown distribution $p_{\mathbf{x},\mathbf{r},\mathbf{q}}$, where $\mathbf{x} \in \mathbb{R}^n$ represents a vector of gene expression values; n is the number of genes; and $\mathbf{r} \in \mathbb{R}^k$ and $\mathbf{q} \in \mathbb{N}^c$ are vectors of k quantitative (e.g. age) and c categorical covariates (e.g. tissue type or gender), respectively. Our goal is to produce realistic gene expression samples by modelling the conditional probability distribution $p(\mathbf{X} = \mathbf{x}|\mathbf{R} = \mathbf{r}, \mathbf{Q} = \mathbf{q})$. By modelling this distribution, we can sample data for different conditions and quantify the uncertainty of the generated expression values.

3.1.2 Adversarial model

Our method builds on a Wasserstein GAN with gradient penalty (WGAN-GP; [127, 18]). Similar to Generative Adversarial Networks (GAN; [111]), WGAN-GPs estimate a generative model via an adversarial process driven by the competition between two players, the *generator* and the *critic*.

Generator The generator aims at producing samples from the conditional $p(\mathbf{X}|\mathbf{R}, \mathbf{Q})$. Formally, we define the generator as a function $G_\theta : \mathbb{R}^u \times \mathbb{R}^k \times \mathbb{N}^c \to \mathbb{R}^n$ parametrised by θ that generates gene expression values $\hat{\mathbf{x}}$ as follows:

$$\hat{\mathbf{x}} = G_\theta(\mathbf{z}, \mathbf{r}, \mathbf{q}), \tag{3.1}$$

where $\mathbf{z} \in \mathbb{R}^u$ is a vector sampled from a fixed noise distribution $p_\mathbf{z}$ and u is a user-definable hyperparameter.

Critic The critic takes gene expression samples $\bar{\mathbf{x}}$ from two input streams (the generator and the data distribution) and attempts to distinguish the true input source. Formally, the critic is a function $D_\omega : \mathbb{R}^n \times \mathbb{R}^k \times \mathbb{N}^c \to \mathbb{R}$ parametrised by ω that we define as follows:

$$\bar{y} = D_\omega(\bar{\mathbf{x}}, \mathbf{r}, \mathbf{q}),$$

where the output $\bar{y}$ is an unbounded scalar that quantifies the degree of realism of an input sample $\bar{\mathbf{x}}$ given the covariates $\mathbf{r}$ and $\mathbf{q}$ (e.g. high values correspond to real samples and low values correspond to fake samples).

Optimisation We optimise the generator and the critic adversarially. Following [127], we train the generator G_θ and the critic D_ω to solve the following minimax game

based on the Wasserstein distance:

$$\begin{aligned} &\min_{\theta}\max_{\omega} \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{r},\mathbf{q}\sim \mathrm{p}_{\mathbf{x},\mathbf{r},\mathbf{q}}} \Big[D_\omega(\mathbf{x},\mathbf{r},\mathbf{q}) - \mathop{\mathbb{E}}_{\mathbf{z}\sim \mathrm{p}_{\mathbf{z}}} [D_\omega(\hat{\mathbf{x}},\mathbf{r},\mathbf{q})] \Big] \\ &\text{subject to} \qquad ||D_\omega(\mathbf{x}_i,\mathbf{r},\mathbf{q}) - D_\omega(\mathbf{x}_j,\mathbf{r},\mathbf{q})|| \leq ||\mathbf{x}_i - \mathbf{x}_j|| \\ &\qquad\qquad\qquad \forall \mathbf{x}_i,\mathbf{x}_j \in \mathbb{R}^n, \mathbf{r}\in\mathbb{R}^k, \mathbf{q}\in\mathbb{N}^c, \end{aligned} \tag{3.2}$$

where $\hat{\mathbf{x}}$ is defined as in Equation 3.1 and the constraint enforces the critic D_ω to be 1-Lipschitz, that is, the norm of the critic's gradient with respect to $\mathbf{x}$ must be at most 1 everywhere.

Let $\{(\mathbf{x}_i,\mathbf{r}_i,\mathbf{q}_i)\}_{i=1}^k$ be a mini-batch of k independent samples from the training dataset $\mathcal{D}$. Let $\{\mathbf{z}_1,\mathbf{z}_2,...,\mathbf{z}_k\}$ be a set of k vectors sampled independently from the noise distribution $p_{\mathbf{z}}$ and let us define the synthetic samples corresponding to the mini-batch as $\hat{\mathbf{x}}_i = G_\theta(\mathbf{z}_i,\mathbf{r}_i,\mathbf{q}_i)$ for each i in $[1,2,...,k]$. We solve the minimax problem described in Equation 3.2 by interleaving mini-batch gradient updates for the generator and the critic, optimising the following problems:

$$\begin{aligned} &\text{Generator:} \quad \min_{\theta} \quad -\frac{1}{k}\sum_{i=1}^{k} D_\omega\Big(\hat{\mathbf{x}}_i,\mathbf{r}_i,\mathbf{q}_i\Big) \\ &\text{Critic:} \quad \min_{\omega} \quad \frac{1}{k}\sum_{i=1}^{k} D_\omega\Big(\hat{\mathbf{x}}_i,\mathbf{r}_i,\mathbf{q}_i\Big) - D_\omega(\mathbf{x}_i,\mathbf{r}_i,\mathbf{q}_i) \\ &\qquad\qquad\qquad + \frac{\lambda}{k}\sum_{i=1}^{k}\Big(||\nabla_{\tilde{\mathbf{x}}_i} D_\omega(\tilde{\mathbf{x}}_i,\mathbf{r}_i,\mathbf{q}_i)||_2 - 1\Big)^2, \end{aligned} \tag{3.3}$$

where λ is a user-definable hyperparameter and each $\tilde{\mathbf{x}}_i$ is a random point along the straight line that connects $\mathbf{x}_i$ and $\hat{\mathbf{x}}_i$, that is, $\tilde{\mathbf{x}}_i = \alpha_i\mathbf{x}_i + (1-\alpha_i)\hat{\mathbf{x}}_i$ with $\alpha_i \sim \mathcal{U}(0,1)$. Intuitively, since enforcing the 1-Lipschitz constraint everywhere (Equation 3.2) is intractable [128], the second term of the critic problem is a relaxed version of the constraint that penalises the gradient norm along points in the straight lines that connect real and synthetic samples [18].

Architecture Figure 3.1 shows the architecture of both players. The generator G receives a noise vector $\mathbf{z}$ as input (green box) as well as sample covariates $\mathbf{r}$ and $\mathbf{q}$ (orange boxes) and produces a vector $\hat{\mathbf{x}}$ of synthetic expression values (red box). The critic D takes either a real gene expression sample $\mathbf{x}$ (blue box) or a synthetic sample $\hat{\mathbf{x}}$ (red box), in addition to sample covariates $\mathbf{r}$ and $\mathbf{q}$, and attempts to distinguish whether the input sample is real or fake. For both players, we use word embeddings [129] to model the sample covariates (light green boxes), a distinctive feature that

allows to learn distributed, dense representations for the different tissue types and, more generally, for all the categorical covariates $\mathbf{q} \in \mathbb{N}^c$.

Formally, let q_j be a categorical covariate (e.g. tissue type) with vocabulary size v_j, that is, $q_j \in \{1, 2, ..., v_j\}$, where each value in the vocabulary $\{1, 2, ..., v_j\}$ represents a different category (e.g. lung or kidney). Let $\bar{\mathbf{q}}_j \in \{0, 1\}^{v_j}$ be a one-hot vector such that $\bar{q}_{jk} = 1$ if $q_j = k$ and $\bar{q}_{jk} = 0$ otherwise. Let d_j be the dimensionality of the embeddings for covariate j. We obtain a vector of embeddings $\mathbf{e}_j \in \mathbb{R}^{d_j}$ as follows:

$$\mathbf{e}_j = \mathbf{W}_j \bar{\mathbf{q}}_j,$$

where each $\mathbf{W}_j \in \mathbb{R}^{d_j \times v_j}$ is a matrix of learnable weights. Essentially, this operation describes a lookup search in a dictionary with v_j entries, where each entry contains a learnable d_j-dimensional vector of embeddings that characterises each of the possible values that q_j can take. To obtain a global collection of embeddings $\mathbf{e}$, we concatenate all the vectors $\mathbf{e}_j$ for each categorical covariate j:

$$\mathbf{e} = \Big\|_{j=1}^{c} \mathbf{e}_j,$$

where c is the number of categorical covariates and $\|$ represents the concatenation operator. We then use the learnable embeddings $\mathbf{e}$ in downstream tasks.

In terms of the player's architecture, we model both the generator G and critic D as neural networks that leverage independent instances $\mathbf{e}^G$ and $\mathbf{e}^D$ of the categorical embeddings for their corresponding downstream tasks. Specifically, we model the two players as follows:

$$G_\theta(\mathbf{z}, \mathbf{r}, \mathbf{q}) = \text{MLP}(\mathbf{z} \| \mathbf{r} \| \mathbf{e}^G) \qquad D_\omega(\bar{\mathbf{x}}, \mathbf{r}, \mathbf{q}) = \text{MLP}(\bar{\mathbf{x}} \| \mathbf{r} \| \mathbf{e}^D),$$

where MLP denotes a multilayer perceptron.

3.1.3 Evaluation metrics

Assessing to what extent simulators are able to generate realistic datasets is a challenging task since we often lack reliable gold standards. Furthermore, unlike for other domains such as image generation, wherein one can empirically assess whether an image is realistic, we do not have an intuitive understanding of high-dimensional transcriptomics data. In order to evaluate the quality of the synthetic data, in this section we propose

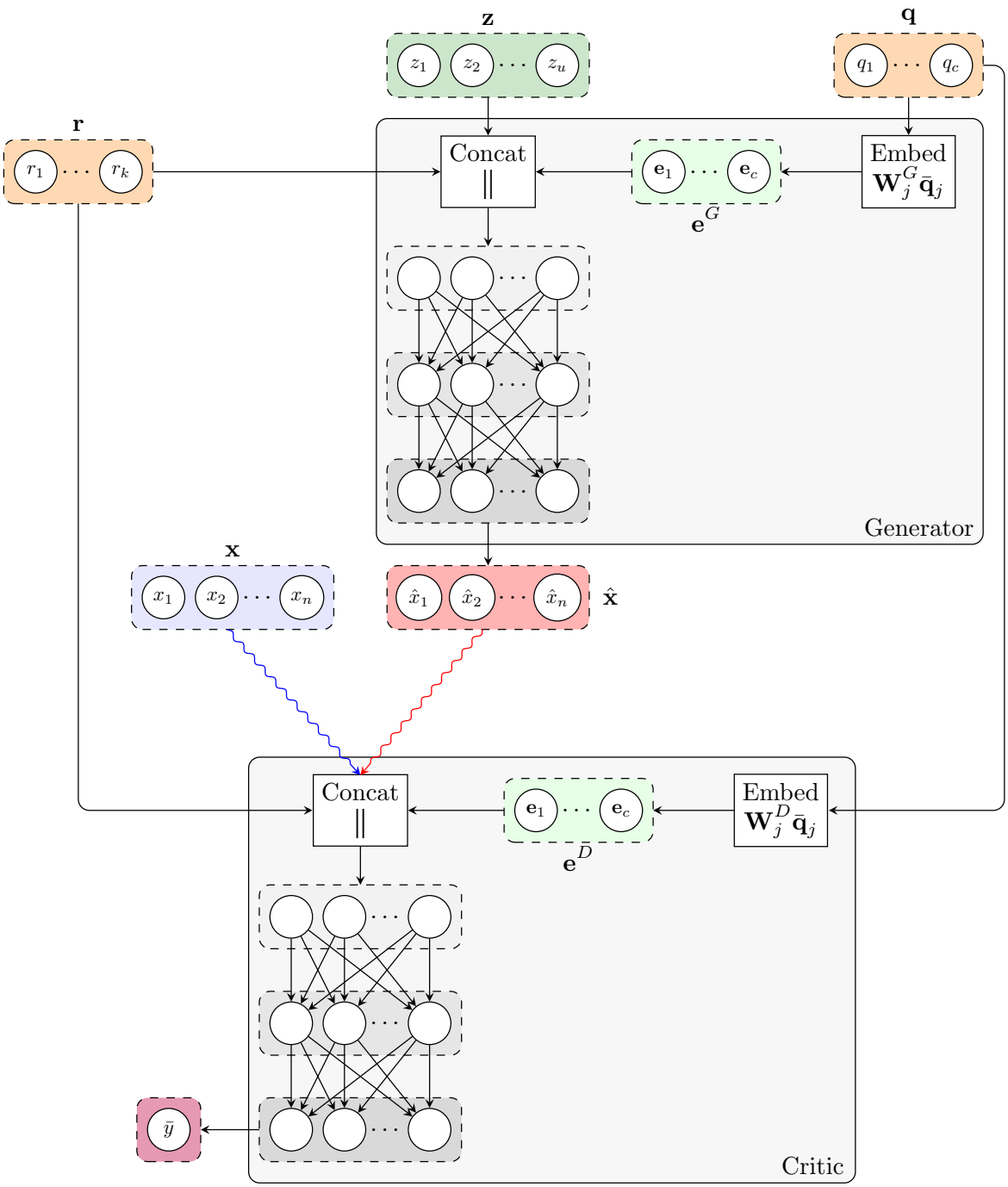


Fig. 3.1 Architecture of our model. The generator receives a noise vector $\mathbf{z}$, and categorical (e.g. tissue type; $\mathbf{q}$) and numerical (e.g. age; $\mathbf{r}$) covariates, and outputs a vector of synthetic expression values ($\hat{\mathbf{x}}$). The critic receives gene expression values from two input streams (real, blue; and synthetic, red) along with the numerical $\mathbf{r}$ and categorical $\mathbf{q}$ covariates, and produces an unbounded scalar $\bar{y}$ that quantifies the degree of realism of the input samples from the two input streams. A characteristic feature of our architecture is the use of word embeddings $\mathbf{e}^G$ and $\mathbf{e}^D$ (green boxes) to learn distributed representations of the categorical covariates for both the generator and the critic.

various quality assessment measures that summarise several statistical properties of gene expression.

We first define a similarity coefficient based on the Pearson's correlation coefficient, which we later use to implement the proposed metrics. Let $\mathbf{A}$ be a $n \times n$ symmetric matrix holding the pairwise distances between all genes. In order to measure how faithfully this matrix preserves the pairwise distances with respect to another $n \times n$ distance matrix $\mathbf{B}$, we define the Pearson's correlation coefficient between the elements in the upper-diagonal of $\mathbf{A}$ and $\mathbf{B}$:

$$\gamma(\mathbf{A}, \mathbf{B}) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(\frac{A_{i,j} - \mu(\mathbf{A})}{\sigma(\mathbf{A})} \right) \left(\frac{B_{i,j} - \mu(\mathbf{B})}{\sigma(\mathbf{B})} \right),$$

where, for a given $n \times n$ matrix $\mathbf{G}$, $\mu(\mathbf{G})$ and $\sigma(\mathbf{G})$ are defined as:

$$\mu(\mathbf{G}) = \frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} G_{i,j}$$

$$\sigma(\mathbf{G}) = \sqrt{\frac{2}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} (G_{i,j} - \mu(\mathbf{G}))^2}$$

General metrics

We first define generic metrics that can be used for any dataset.

Distance between *real* and *artificial* distance matrices (S_{dist}) Let $\mathbf{X} \in \mathbb{R}^{m_1 \times n}$ and $\mathbf{Z} \in \mathbb{R}^{m_2 \times n}$ be two matrices containing m_1 real and m_2 synthetic observations for n genes, respectively. For a given distance function d, we define two $n \times n$ distance matrices $\mathbf{D}^X$ and $\mathbf{D}^Z$ as:

$$D^X_{i,j} = d(col(\mathbf{X}, i), col(\mathbf{X}, j)) \quad D^Z_{i,j} = d(col(\mathbf{Z}, i), col(\mathbf{Z}, j)), \tag{3.4}$$

where $col(\mathbf{X}, i)$ is the i-th column of matrix $\mathbf{X}$. Throughout the remainder of the chapter we use the Pearson's dissimilarity coefficient as the distance function d.

The coefficient $S_{\text{dist}} = \gamma(\mathbf{D}^X, \mathbf{D}^Z)$ measures whether the pairwise distances between genes from the real data are correlated with those from the synthetic data.

Distance between *real* and *artificial* dendrograms (S_{dend}) Let $C : \mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$ be a function that performs agglomerative hierachical clustering according to a

given linkage function, taking a $n \times n$ distance matrix as input and returning the $n \times n$ distance matrix of the resulting dendrogram. Intuitively, each element (i, j) in the dendrogrammatic distance matrices measures the distance between the two outermost clusters that separate genes i and j.

The coefficient $S_{\text{dend}} = \gamma(C(\mathbf{D}^X), C(\mathbf{D}^Z))$ measures the structural similarity between the dendrograms, giving a score close to 1 when the *real* and *artificial* dendrograms have a similar structure. Consequently, this metric encourages the synthetic distribution to preserve the relationships among groups of genes that are found in the real distribution. Importantly, this coefficient does not necessarily correlate with $\gamma(\mathbf{D}^X, \mathbf{D}^Z)$ (Supplementary Information B.1).

GRN-specific metrics

The following metrics make use of an a priori known GRN to evaluate statistical properties of gene regulatory interactions.

Weighted sum of TF-TG similarity coefficients ($S_{\text{TF-TG}}$) Let $\mathcal{G}$ be a function returning the set of indices of the target genes (TGs) that are regulated by a given transcription factor (TF). For a given dataset $\mathbf{D}$ and a TF f, let $\mathbf{r}_f^D$ be a vector of distances between the expressions of f and the expressions of its target genes:

$$\mathbf{r}_f^D = \Big(d(col(\mathbf{D}, f), col(\mathbf{D}, g)) : g \in \mathcal{G}(f)\Big)^\top,$$

where d is an arbitrary distance measure. If the synthetic dataset $\mathbf{Z}$ is realistic with respect to the real dataset $\mathbf{X}$, the vectors $\mathbf{r}_f^X$ and $\mathbf{r}_f^Z$ will be similar for each TF f in a set of transcription factors $\mathcal{F}$. Let w_f be a coefficient associated with the importance of TF f (e.g. we choose $w_f = |\mathcal{G}(f)|$). We summarise this information as follows:

$$S_{\text{TF-TG}}(\mathbf{X}, \mathbf{Z}) = \frac{1}{\sum_{f \in \mathcal{F}} w_f} \sum_{f \in \mathcal{F}} w_f \cdot \upsilon(\mathbf{r}_f^X, \mathbf{r}_f^Z),$$

where $\upsilon(\mathbf{r}_f^X, \mathbf{r}_f^Z)$ is the cosine similarity between vectors $\mathbf{r}_f^X$ and $\mathbf{r}_f^Z$. The coefficient $S_{\text{TF-TG}}(\mathbf{X}, \mathbf{Z})$ measures whether the TF-TG dependencies in the synthetic data resemble those from the real data.

Weighted sum of TG-TG similarity coefficients ($S_{\text{TG-TG}}$) Similarly, we define a coefficient $S_{\text{TG-TG}}$ to measure whether the expression of TGs regulated by the same

TF in synthetic data conforms well with the analog expressions in real data:

$$S_{\mathrm{TG-TG}}(\mathbf{X},\mathbf{Z}) = \frac{1}{\sum_{f\in\mathcal{F}} w_f} \sum_{f\in\mathcal{F}} w_f \sum_{g\in\mathcal{G}(f)} \upsilon(\mathbf{q}^X_{f,g}, \mathbf{q}^Z_{f,g}),$$

where, for a given matrix $\mathbf{G}$, $\mathbf{q}^G_{f,g}$ is the vector of distances between gene g and all the genes regulated by f (excluding g):

$$\mathbf{q}^G_{f,g} = \Big(d(col(\mathbf{G},g), col(\mathbf{G},i)) : i \in (\mathcal{G}(f) - \{g\})\Big)^\top$$

3.2 Results

Here we assess the quality of the synthetic data produced by our generative model. We compared our approach to existing simulators of gene expression, including GeneNetWeaver [124] and SynTReN [123], evaluating several properties of gene expression using an *E. coli* dataset. We then studied the ability of our approach to produce tissue-specific gene expression for several cancer and healthy human tissues.

3.2.1 *E. coli* evaluation

To analyse to what extent the proposed generative model is able to preserve statistical properties of gene regulatory interactions, we introduce a first case study that leverages *E. coli* transcriptomics data from the M^{3D} database [130]. We chose this bacterium because it has a relatively simple genome (∼4,400 genes) and its gene expression mechanisms are well understood [131] and characterised by the RegulonDB database [126]. In particular, we selected a meaningful subset of *E. coli* genes whose expression is directly or indirectly regulated by the master regulator cAMP receptor protein (CRP).

Many Microbe Microarrays Database We downloaded *E. coli* single-channel Affymetrix microarray data from the Many Microbe Microarrays Database (M^{3D}; [130]). From the 7459 available probes, we excluded those corresponding to intergenic regions and controls, resulting in a dataset of 907 samples and 4297 features. These probes were uniformly normalised by [130] using log-scale robust multi-array average (RMA; [118]) to reduce batch effects and make the samples comparable across conditions. To scale the data, we applied z-score normalisation for every gene.

RegulonDB The gene regulatory network of *E. coli* is one of the most well-characterised transcriptional networks of a single cell. RegulonDB [126] is a database

that integrates biological knowledge about the transcriptional regulatory mechanisms of *E. coli*. The database gathers information from multiple biological studies to reconstruct the structure of the *E. coli* gene regulatory network. We leveraged information from RegulonDB to select the subnetwork of genes corresponding to the cAMP receptor protein (CRP) regulatory hierarchy, allowing us to study whether regulatory associations are preserved in the in-silico-generated data.

CRP hierarchy To reduce the dimensionality of the dataset and enable learning from a scarce number of samples, we performed breadth-first search on the RegulonDB interactions to select a meaningful subset of genes whose expression is directly or indirectly regulated by cAMP receptor protein (CRP). We broke loops by removing non-tree edges as we built the hierarchy. The cAMP receptor protein, which regulates global patterns of transcription in response to carbon availability, is one of the best characterised global transcriptional regulators in *E. coli* [131].

Baselines We compared our approaches with other existing methods: SynTReN [123] and GeneNetWeaver (GNW; [124]). Given a gene regulatory network, these two methods model regulatory interactions with ordinary and stochastic differential equations based on Michaelis-Menten and Hill kinetics. These two models have been widely used to produce synthetic gene expression data from gene regulatory networks with the purpose of benchmarking network inference algorithms, but they have been previously criticised because they fail to emulate key properties of gene expression [17]. For example, [17] showed that clustering genes according to gene expression yields clusters that are significantly different to those of real data, and that the correlations between transcription factors and target genes are poorly preserved.

We generated a gene expression dataset of 680 samples using our generative model, SynTReN, and GNW. For SynTReN and GNW, we created a network with 1076 nodes (without background nodes; e.g. external nodes that regulate the expression of genes in the network) corresponding to the CRP hierarchy. In both cases, we selected the configuration that optimises the S_{dist} score. For SynTReN, this corresponded to a biological noise level of 0.8 out of 1 and an experimental noise level of 0 (Supplementary Information B.2). For GNW, the best coefficient for the noise term of the stochastic differential equations was 0.1 (Supplementary Information B.3).

Statistical properties of regulatory interactions Table 3.1 shows a quantitative comparison of the three methods. We determined an approximate lower bound on the

Table 3.1 Quantitative assessment of the generated data with results for a *random* and a *real* (M^{3D} train) simulators.

Simulator	S_{dist}	S_{dend}	$S_{\text{TF-TG}}$	$S_{\text{TG-TG}}$
Random	0.0000	-0.0002	0.2299	-0.0132
Real	0.9109	0.5197	0.9143	0.9467
SynTReN	0.0449	0.0444	0.2134	0.2594
GNW	0.0587	0.0223	0.1838	0.1930
GAN	**0.8145**	**0.3872**	**0.8386**	**0.8734**

metrics by randomly generating gene expression data following a uniform distribution $\mathcal{U}(0,1)$. We determined an approximate upper bound by using the real *E. coli* gene expression samples in the train dataset. The proposed model closely approximated the upper bound in every metric, outperforming SynTReN and GNW by a large margin. In fact, SynTReN and GNW performed similarly to the random simulator. We attribute this to the fact that SynTReN and GNW rely exclusively on the source GRNs to produce synthetic data. In contrast, our proposed WGAN-GP model leverages real expression data to optimise a generative model in an unsupervised manner without requiring information on the regulatory interactions. In Supplementary Information B.4, we further analysed differences between the three simulators in terms of the distributions proposed by [17]. Overall, our results show that the synthetic data faithfully preserves key properties of gene expression, such as correlations between the expression of transcription factors and their target genes, and demonstrate the generality and application of the method in bacterial populations.

3.2.2 Generating tissue-specific human transcriptomic data

We introduce a second case study to analyse the ability of the proposed method to generate human RNA-seq data from a broad range of cancer and normal tissue types. Specifically, we combined data from GTEx and TCGA, two reference resources for the scientific community that have generated a comprehensive collection of human transcriptome data in a diverse set of tissues and cancer types.

The Genotype-Tissue Expression dataset The Genotype-Tissue Expression (GTEx) dataset collected transcriptomics data of multiple tissues from around 838 human donors [12] (healthy individuals). The biospecimen repository includes model systems such as whole blood and Epstein Barr virus (EBV) transformed lymphocytes;

central nervous system tissues from 13 brain regions; and a wide diversity of other primary tissues from *non-diseased* individuals.

The Cancer Genome Atlas The Cancer Genome Atlas (TCGA) is a public database that aims to increase the understanding of the genetic basis of a wide range of cancers. The biospecimen repository includes high-throughput genomic data from *diseased* and matched *healthy* samples spanning 33 cancer types [132].

Data integration We specifically selected samples from 15 common tissues in GTEx and TCGA, namely lung, breast, kidney, thyroid, colon, stomach, prostate, salivary, liver, esophagus muscularis, esophagus mucosa, esophagus gastrointestinal, bladder, uterus, and cervix. To unify the data and correct for batch effects, we followed the pipeline described by [84]. After integrating the data, our dataset consisted of 9147 samples and 18154 genes. We trained our WGAN-GP model on the combined RNAseqDB [84] (GTEx+TCGA) dataset and sampled a synthetic dataset that matched the test set both in number of samples (2287) and proportions of tissue and cancer types.

Correlation and cluster analysis Figure 3.2 shows the pairwise correlations and dendrograms for 14 important cancer driver genes with high mutation frequency [133]. For this subset of genes, our model closely matched the correlation and clustering expression patterns of the real data. To evaluate the clustering quality at a larger scale, we applied k-means to both the test and the generated expression datasets (Figure 3.3). We observed a bijective mapping between real and synthetic gene clusters. In other words, for each real cluster, there was a synthetic cluster that shared the majority of genes (and vice-versa). We further performed over-representation analysis with GOfuncR [134]. We noted that similar Gene Ontology terms were enriched for each matching pair of gene clusters. Using the real test set as the reference dataset, we computed the metrics from Section 3.1.3. We quantified S_{dist} at 0.920 out of 0.947 and S_{dend} at 0.215 out of 0.222, where the bounds were approximate and given by the metrics applied to the train set. These results suggest that the generated data retains local and global co-expression patterns.

Over-representation analysis We review the intuitions behind over-representation analysis in Chapter 2 (Section 2.2.2).

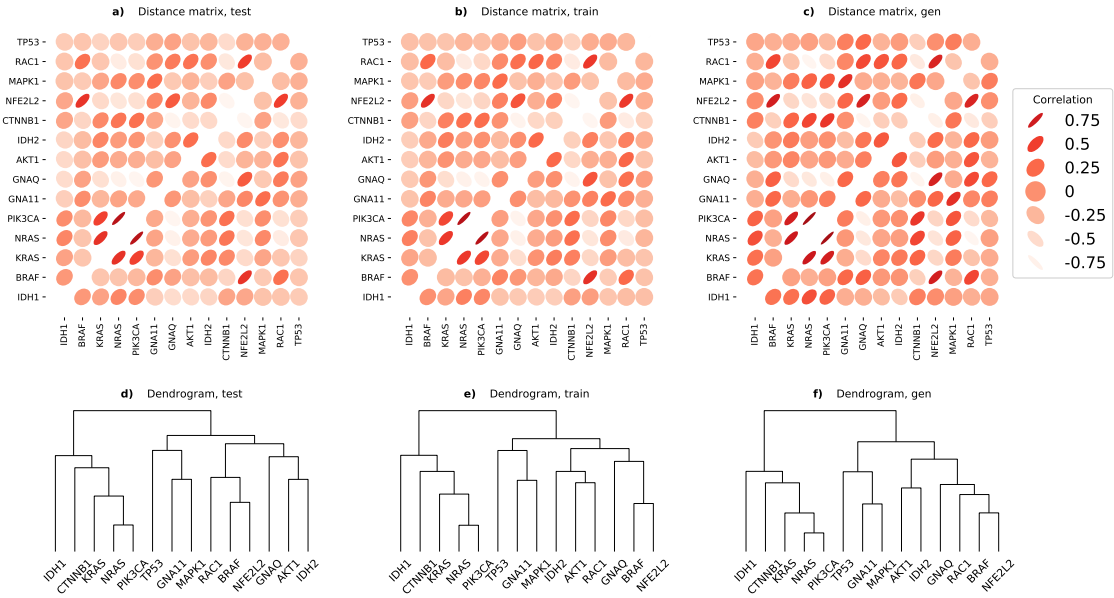


Fig. 3.2 Correlation matrices and dendrograms for a subset of 14 cancer driver genes with high mutation frequency [133]. (a, b, c) Correlation matrices computed using the 2287 test set (unseen during training), 6860 train set, and 2287 in-silico generated samples from the test set, respectively. For the synthetic data, the distribution of gene correlations was slightly flatter (Supplementary Information B.4). (d, e, f) Dendrograms obtained by performing hierachical clustering with complete linkage on the same datasets. Our in-silico generated data closely matched the expression patterns in terms of gene correlations and clusters.

Tissue and cancer-specific gene expression traits Next, we tested whether the synthetic data accounts for tissue-specific and cancer-specific traits of gene expression. We generated a gene expression dataset matching the statistics of the train set (i.e. size and proportions of tissue and cancer types) and used the synthetic data to train a multilayer perceptron (MLP; 2 hidden layers of 64 units with ReLU activations) to perform tissue and cancer type classification. For tissue type classification (15 tissues), the scores for the MLP trained on the synthetic data were $AUC = 0.9884 \pm 0.0010$ and $F1 = 0.9222 \pm 0.0040$ (real test set; averaged over 5 runs). The same figures for the MLP trained on real data were $AUC = 0.9986 \pm 0.0003$ and $F1 = 0.9860 \pm 0.0007$. For cancer-normal binary classification, the scores were $AUC = 0.9992 \pm 0.0001$ and $F1 = 0.9893 \pm 0.0009$ for the MLP trained on synthetic data, and $AUC = 0.9997 \pm 0.0001$ and $F1 = 0.9939 \pm 0.0005$ for the MLP trained on real data. We then analysed the expression manifold using UMAP [135] and observed a complete overlap of the real

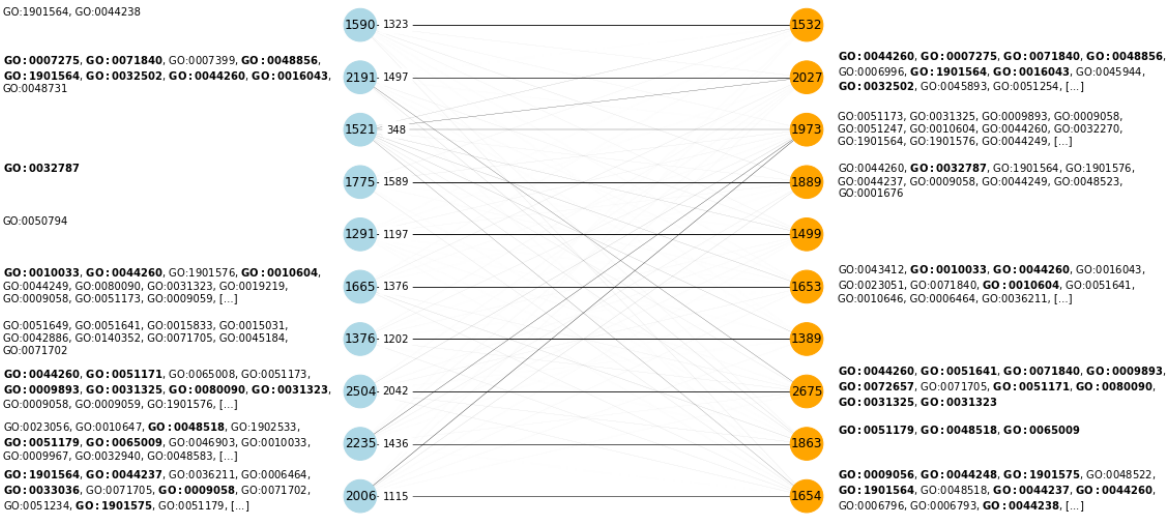


Fig. 3.3 Cluster analysis on the real and synthetic expression datasets. We performed k-means clustering with k=10 clusters on the test (real) and generated datasets. Blue and orange nodes represent real and synthetic clusters, respectively. The value of each node corresponds to the number of genes in that cluster. We matched real and synthetic clusters according to the number of shared genes and displayed the number of matching genes in the edge labels for the top associations. The width of each edge is proportional to the number of shared genes. We further performed an over-representation test using GOfuncR [134] with a family-wise error rate (FWER) threshold of 0.05. We show the enriched Gene Ontology terms next to the corresponding cluster and highlight in bold those that are common between each top matching pair of clusters (see Supplementary Information B.5 for a detailed list of the enriched Gene Ontology terms). These results suggest that gene clusters and enriched biological processes were similar at a global scale.

and synthetic samples (Figure 3.4). The UMAP representation revealed strong clusters of gene expression data across a variety of normal and cancer tissues. Overall, these results show that our method can emulate tissue- and disease-specific traits of gene expression.

Candidate causal biomarkers of cancer types Our model allows producing gene expression data for synthetic individuals across different tissues and cancer types. The gene expression data of each donor is fully determined by a latent vector and a set of covariates (e.g. tissue-type and cancer-type). If we clamp the latent variable and covariates to a fixed value, we can then use the generator to produce gene expression data for the same *counterfactual* individual with and without cancer. Changes in gene

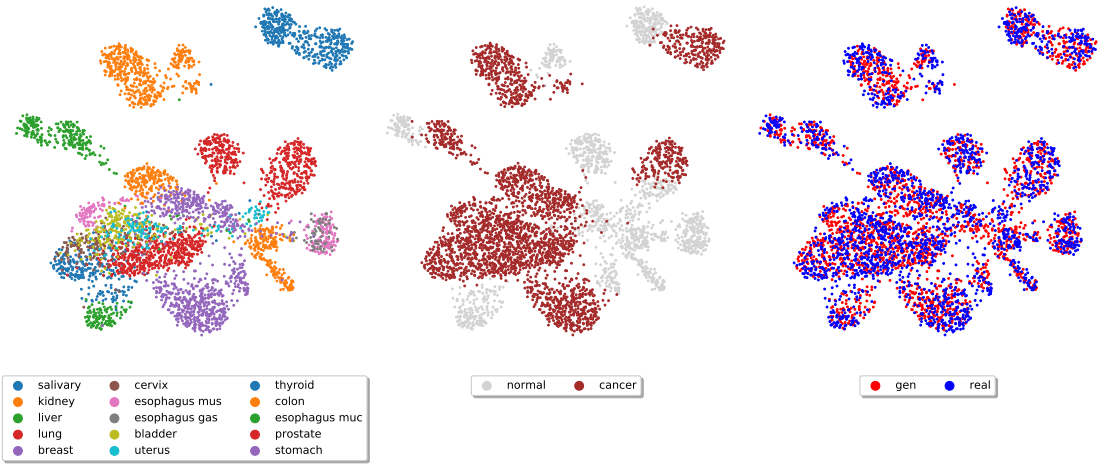


Fig. 3.4 UMAP representation of RNA-seq data across 15 tissue types for both normal and cancer, combining data from the test set (2287 samples) and synthetic data (2287 samples). The first plot is colored by tissues, the second indicates which samples are carcinogenic, and the third distinguishes samples between real and synthetic. Our generative model can produce realistic data for different tissues and diseases.

expression will then be associated with the cancer factor, as all other latent variables and covariates are kept constant across counterfactual samples. This cannot be done for the real transcriptomics data because we do not have access to counterfactuals and, therefore, changes in gene expression between healthy and cancer donors might be associated with a large number of confounders in addition to cancer. Other works have explored this idea in the context of image editing [136–138].

To rank the genes according to their sensitivity to cancer in our model, we generated pairs of counterfactual gene expression values in several tissues. For each pair of measurements, we fixed all the latent variables to the same state and produced healthy and cancerous gene expression. We then computed the differential expression values and averaged the results across 1000 runs, obtaining differential gene expression signatures for each cancer type. Finally, we ranked the genes separately for each cancer type and reported the resulting ranking in Table 3.2, along with literature references for each reported gene. The resulting candidate genes warrant further investigations because their expression changes were associated with cancer in an unconfounded way, i.e. all the other determinants of expression in the model were kept fixed. Importantly, the gene ranking is sensitive to the ability of our model to estimate the joint probability distribution of gene expression conditioned on the covariates.

Table 3.2 Candidate causal biomarkers for different cancer types. To generate these results, we clamped the WGAN-GP latent variables and covariates to a fixed value and used the generator to produce counterfactual healthy and cancer gene expression for each individual. We then computed differential expression values and averaged the results across 1000 runs, obtaining cancer-type-specific signatures. We finally ranked genes separately for each cancer type and included supporting references for the top genes. Importantly, these results are sensitive to the ability of our model to estimate the probability distribution of gene expression conditioned on the covariates.

Cancer-type	Sign	Top 5 genes	References
Colon	+	WNK4	[139, 140]
	+	TMEM35	[141, 142]
	+	AGR3	[143, 144]
	–	NSA2	
	+	TOMM34	[145–147]
Breast	+	RP11-318A15.7	
	+	KLF2	[148, 149]
	+	PCDH19	[150, 151]
	+	VIP	[152–154]
	+	CYP2U1	[155, 156]
Thyroid	–	MCM3	[157, 158]
	–	SYK	[159]
	+	TBCCD1	[160, 161]
	+	ZBBX	
	+	MESDC1	[162, 163]
Prostate	+	MTRNR2L8	[164]
	+	FAM204A	
	+	CASP3	[165–167]
	–	HABP2	[168, 169]
	+	PIF1	[170]

3.3 Discussion

In this chapter, we implemented a simulator based on a Wasserstein Generative Adversarial Network with gradient penalty [18]. We studied the problem of generating realistic transcriptomics data and analysed several statistical properties of gene expression in two case studies: *E. coli* microarray data and human RNA-seq data across a broad range of tissue and cancer types.

For the first case study, we compared the ability of our simulator to preserve gene expression properties related to the underlying gene regulatory network of the organism, e.g. *E. coli*. Importantly, we noted that two widely used simulators, SynTReN and

GeneNetWeaver (GNW), poorly preserve correlation properties of gene expression, such as TF-TG and TG-TG correlations. This has important implications on the benchmarks of algorithms that reverse engineer the GRN from transcriptomics data. In particular, if these correlations are not well-preserved, it is not possible to guarantee the generalisability of such algorithms to real data. We showed that the data produced by our model is highly realistic according to these metrics, outperforming SynTReN and GNW by a large margin.

For the second case study, we trained our model on a dataset that combines RNA-seq data from the GTEx and TCGA projects. Our analysis showed that the proposed approach preserves correlation and clustering properties, suggesting that the model learns to approximate the gene expression manifold in a biologically meaningful way. Furthermore, our model seems to capture tissue- and cancer-specific properties of transcriptomic data. Finally, we proposed a tool based on the simulator that might be employed by researchers to explore *candidate* cancer driver genes, with potential application in biomarker discovery.

Chapter 4

Intra-tissue imputation of gene expression

High-throughput profiling of the transcriptome has revolutionised discovery methods in the biological sciences. The resulting gene expression measurements can be used to uncover disease mechanisms [171–173], propose novel drug targets [174, 175], provide a basis for comparative genomics [176, 177], and motivate a wide range of fundamental biological problems. In parallel, methods that learn to represent the expression manifold can improve our mechanistic understanding of complex traits, with potential methodological and technological applications, including organs-on-chips [178] and synthetic biology [179], and the integration of heterogeneous transcriptomics datasets.

A question of fundamental biological significance is to what extent the expression of a subset of genes can be used to recover the full transcriptome with minimal reconstruction error. Genes that participate in similar biological processes or that have shared molecular function are likely to have similar expression profiles [19], prompting the question of gene expression prediction from a minimal subset of genes. Moreover, gene expression measurements may suffer from unreliable values because some regions of the genome are extremely challenging to interrogate due to high genomic complexity or sequence homology [20], further highlighting the need for accurate imputation. Moreover, most gene expression studies continue to be performed with specimens derived from peripheral blood or a convenient surrogate (e.g., lymphoblastoid cell lines; LCLs) due to the difficulty of collecting some tissues. However, gene expression may be tissue or cell-type specific, potentially limiting the utility of a proxy tissue.

The research presented in this chapter has been conducted in collaboration with Tiago Azevedo, Eric R. Gamazon, and Pietro Liò

The missing data problem can adversely affect downstream gene expression analysis. The simple approach of excluding samples with missing data from the analysis can lead to a substantial loss in statistical power. Dimensionality reduction approaches such as principal component analysis (PCA) and singular value decomposition (SVD) [180] cannot be applied to gene expression data with missing values without a previous imputation step. Clustering methods, a mainstay of genomics, such as k-means and hierarchical clustering may become unstable even with a few missing values [181].

To address these challenges, we develop two deep learning approaches to gene expression imputation: Generative Adversarial Imputation Networks for GTEx (GAIN-GTEx) and Pseudo-Mask Imputer (PMI). In both cases, we present an architecture that recovers missing expression data for multiple tissue types under different levels of missingness. In contrast to existing linear methods for deconfounding gene expression [182], our methods integrate covariates (global determinants of gene expression; [125]) to account for their non-linear effects. In particular, a characteristic feature of our architectures is the use of word embeddings [129] to learn rich and distributed representations for the tissue types and other covariates. To enlarge the possibility and scale of a study's expression data (e.g., by including samples from highly inaccessible tissues), we train our model on RNA-Seq data from the Genotype-Tissue Expression (GTEx) project [48, 183], a reference resource (v8) that has generated a comprehensive collection of human transcriptome data in a diverse set of tissues.

We show that the proposed approaches compare favourably to several standard and state-of-the-art imputation methods in terms of predictive performance and runtime. In performance comparison on the protein-coding genes, GAIN-GTEx outperforms all the other methods in in-place imputation while PMI displays the highest performance in inductive imputation. Furthermore, we demonstrate that our methods are highly applicable across diverse tissues and varying levels of missingness. Finally, to analyse the cross-study relevance of our approach, we perform imputation on gene expression data from The Cancer Genome Atlas (TCGA; [132]) and show that our approach is robust when applied to independent RNA-Seq data. Our code is publicly available at: `https://github.com/rvinas/GTEx-imputation`

4.1 Methodology

In this section, we introduce two deep learning approaches for gene expression imputation with broad applicability, allowing us to investigate their strengths and weaknesses in several realistic scenarios. Throughout the remainder of the chapter, we use script

letters to denote sets (e.g., $\mathcal{D}$), upper-case bold symbols to denote matrices or random variables (e.g., $\mathbf{X}$), and lower-case bold symbols to denote column vectors (e.g., $\mathbf{x}$ or $\bar{\mathbf{q}}_j$). The rest of the symbols (e.g., $\bar{q}_{jk}$, G, or f) denote scalar values or functions.

4.1.1 Problem formulation

Consider a dataset $\mathcal{D} = \{(\tilde{\mathbf{x}}, \mathbf{m}, \mathbf{r}, \mathbf{q})\}$, where $\tilde{\mathbf{x}} \in \mathbb{R}^n$ represents a vector of gene expression values with missing components; $\mathbf{m} \in \{0,1\}^n$ is a mask indicating which components of the original vector of expression values $\mathbf{x}$ are missing or observed; n is the number of genes; and $\mathbf{q} \in \mathbb{N}^c$ and $\mathbf{r} \in \mathbb{R}^k$ are vectors of c categorical (e.g., tissue type or sex) and k quantitative covariates (e.g., age), respectively. Our goal is to recover the original gene expression vector $\mathbf{x} \in \mathbb{R}^n$ by modelling the conditional probability distribution $p(\mathbf{X} = \mathbf{x} | \tilde{\mathbf{X}} = \tilde{\mathbf{x}}, \mathbf{M} = \mathbf{m}, \mathbf{R} = \mathbf{r}, \mathbf{Q} = \mathbf{q})$, where the upper-case symbols denote the corresponding random variables.

4.1.2 Pseudo-mask imputation

We first introduce a novel imputation method named Pseudo-Mask Imputer (PMI).

Formulation Let $\tilde{\mathbf{x}} = \mathbf{m} \odot \mathbf{x} \in \mathbb{R}^n$ be a vector of gene expression values whose missing components are indicated by a mask vector $\mathbf{m} \in \{0,1\}^n$. Our model is a function $f : \mathbb{R}^n \times \{0,1\}^n \times \mathbb{R}^k \times \mathbb{N}^c \to \mathbb{R}^n$ that imputes the missing expression values $(\mathbf{1} - \mathbf{m}) \odot \mathbf{x}$ as follows:

$$\bar{\mathbf{x}} = f(\tilde{\mathbf{x}}, \mathbf{m}, \mathbf{r}, \mathbf{q}).$$

Here $\odot$ denotes element-wise multiplication. The recovered vector of gene expression values is then given by $\mathbf{m} \odot \tilde{\mathbf{x}} + (\mathbf{1} - \mathbf{m}) \odot \bar{\mathbf{x}}$.

Optimisation We optimise the model to maximise the imputation performance on a dynamic subset of observed, *pseudo-missing* components. In particular, we first generate a *pseudo-mask* $\tilde{\mathbf{m}}$ as follows:

$$\tilde{\mathbf{m}} = \mathbf{m} \odot \mathbf{b} \qquad \mathbf{b} \sim B(1, p) \qquad p \sim U(\alpha, \beta),$$

where $\mathbf{b} \in \{0,1\}^n$ is a vector sampled from a Bernoulli distribution B and $\alpha \in [0,1]$ and $\beta \in [\alpha, 1]$ are hyperparameters that parameterise a uniform distribution U. Using the *pseudo-mask* $\tilde{\mathbf{m}}$, we split the observed expression values into a set of *pseudo-observed*

Algorithm 1: Training algorithm

Input: Input dataset $\mathcal{D} = \{(\mathbf{x}, \mathbf{m}, \mathbf{r}, \mathbf{q})\}$, batch size B, hyperparameters α and β

- Initialise parameters of the model f

while *not convergence criteria reached* **do**

- Sample mini-batch:
  $\{(\mathbf{x}^{(i)}, \mathbf{m}^{(i)}, \mathbf{r}^{(i)}, \mathbf{q}^{(i)})\}_{i=1}^{B} \sim \mathcal{D}$
- Sample *pseudo-mask* for each example of the mini-batch:
  $p^{(i)} \sim U(\alpha, \beta)$
  $\mathbf{b}^{(i)} \sim B(1, p^{(i)})$
  $\tilde{\mathbf{m}}^{(i)} = \mathbf{m}^{(i)} \odot \mathbf{b}^{(i)}$
- Split components into *pseudo-observed* and *pseudo-missing*:
  $\tilde{\mathbf{x}}^{(i)} = \mathbf{x}^{(i)} \odot \tilde{\mathbf{m}}^{(i)}$
  $\tilde{\mathbf{y}}^{(i)} = \mathbf{x}^{(i)} \odot \mathbf{m}^{(i)} \odot (\mathbf{1} - \tilde{\mathbf{m}}^{(i)})$
- Impute *pseudo-missing* components:
  $\bar{\mathbf{x}}^{(i)} = f(\tilde{\mathbf{x}}^{(i)}, \tilde{\mathbf{m}}^{(i)}, \mathbf{r}^{(i)}, \mathbf{q}^{(i)})$
- Optimise the model by descending its stochastic gradient:
  $\nabla \frac{1}{B} \sum_{i=1}^{B} \mathcal{L}(\bar{\mathbf{x}}^{(i)}, \tilde{\mathbf{y}}^{(i)}, \mathbf{m}^{(i)}, \tilde{\mathbf{m}}^{(i)})$

end

components $\tilde{\mathbf{x}}$ and a set of *pseudo-missing* components $\tilde{\mathbf{y}}$:

$$\tilde{\mathbf{x}} = \mathbf{x} \odot \tilde{\mathbf{m}} \qquad \tilde{\mathbf{y}} = \mathbf{x} \odot \mathbf{m} \odot (\mathbf{1} - \tilde{\mathbf{m}}),$$

The imputed components are then given by $\bar{\mathbf{x}} = f(\tilde{\mathbf{x}}, \tilde{\mathbf{m}}, \mathbf{r}, \mathbf{q})$. We optimise our model to minimise the mean squared error between the ground truth and the imputed *pseudo-missing* components:

$$\mathcal{L}(\bar{\mathbf{x}}, \tilde{\mathbf{y}}, \mathbf{m}, \tilde{\mathbf{m}}) = \frac{1}{Z}\Big(\mathbf{m} \odot (\mathbf{1} - \tilde{\mathbf{m}})\Big)^{\top} (\bar{\mathbf{x}} - \tilde{\mathbf{y}})^2,$$

where $Z = \Big(\mathbf{m} \odot (\mathbf{1} - \tilde{\mathbf{m}})\Big)^{\top} \Big(\mathbf{m} \odot (\mathbf{1} - \tilde{\mathbf{m}})\Big)$ is a normalisation term. We summarise our training algorithm in Algorithm 2.

Importantly, the *pseudo-mask* mechanism generates different sets of *pseudo-observed* components for each input example, effectively increasing the number of training samples. Specifically, the hyperparameters α and β control the fraction of *pseudo-observed* and *pseudo-missing* components through the probability $p \sim U(\alpha, \beta)$. On one hand, a low probability p yields sparse *pseudo-observed* vectors $\hat{\mathbf{x}}$, resulting in fast convergence but high bias. On the other hand, a high probability p yields denser *pseudo-observed* vectors $\hat{\mathbf{x}}$, resulting in low bias but slower convergence. At inference time, p is set to 1 and the *pseudo-mask* $\tilde{\mathbf{m}}$ is equal to the input mask $\mathbf{m}$.

Architecture We model the imputer f as a neural network. We first describe how we use word embeddings, a distinctive feature that allows learning rich, dense representations for the different tissue types and, more generally, for all the covariates $\mathbf{q} \in \mathbb{N}^c$.

Formally, let q_j be a categorical covariate (e.g., tissue type) with vocabulary size v_j, that is, $q_j \in \{1, 2, ..., v_j\}$, where each value in the vocabulary $\{1, 2, ..., v_j\}$ represents a different category (e.g., whole blood or kidney). Let $\bar{\mathbf{q}}_j \in \{0, 1\}^{v_j}$ be a one-hot vector such that $\bar{q}_{jk} = 1$ if $q_j = k$ and $\bar{q}_{jk} = 0$ otherwise. Let d_j be the dimensionality of the embeddings for covariate j. We obtain a vector of embeddings $\mathbf{e}_j \in \mathbb{R}^{d_j}$ as follows:

$$\mathbf{e}_j = \bar{\mathbf{q}}_j^\top \mathbf{W}_j, \tag{4.1}$$

where each $\mathbf{W}_j \in \mathbb{R}^{v_j \times d_j}$ is a matrix of learnable weights. Essentially, this operation describes a lookup search in a dictionary with v_j entries, where each entry contains a learnable d_j-dimensional vector of embeddings that characterise each of the possible values that q_j can take. To obtain a global collection of embeddings $\mathbf{e}$, we concatenate all the vectors $\mathbf{e}_j$ for each categorical covariate j:

$$\mathbf{e} = \Big\|_{j=1}^{c} \mathbf{e}_j, \tag{4.2}$$

where c is the number of categorical covariates and $\|$ represents the concatenation operator. We then use the learnable embeddings $\mathbf{e}$ in downstream tasks.

In terms of the architecture, we model f as follows:

$$f(\tilde{\mathbf{x}}, \mathbf{m}, \mathbf{r}, \mathbf{q}) = \text{MLP}(\tilde{\mathbf{x}} \| \mathbf{m} \| \mathbf{r} \| \mathbf{e}),$$

where MLP denotes a multilayer perceptron and $\tilde{\mathbf{x}} = \mathbf{x} \odot \mathbf{m} \in \mathbb{R}^n$ is the masked gene expression. Figure 4.1 shows the architecture of the model.

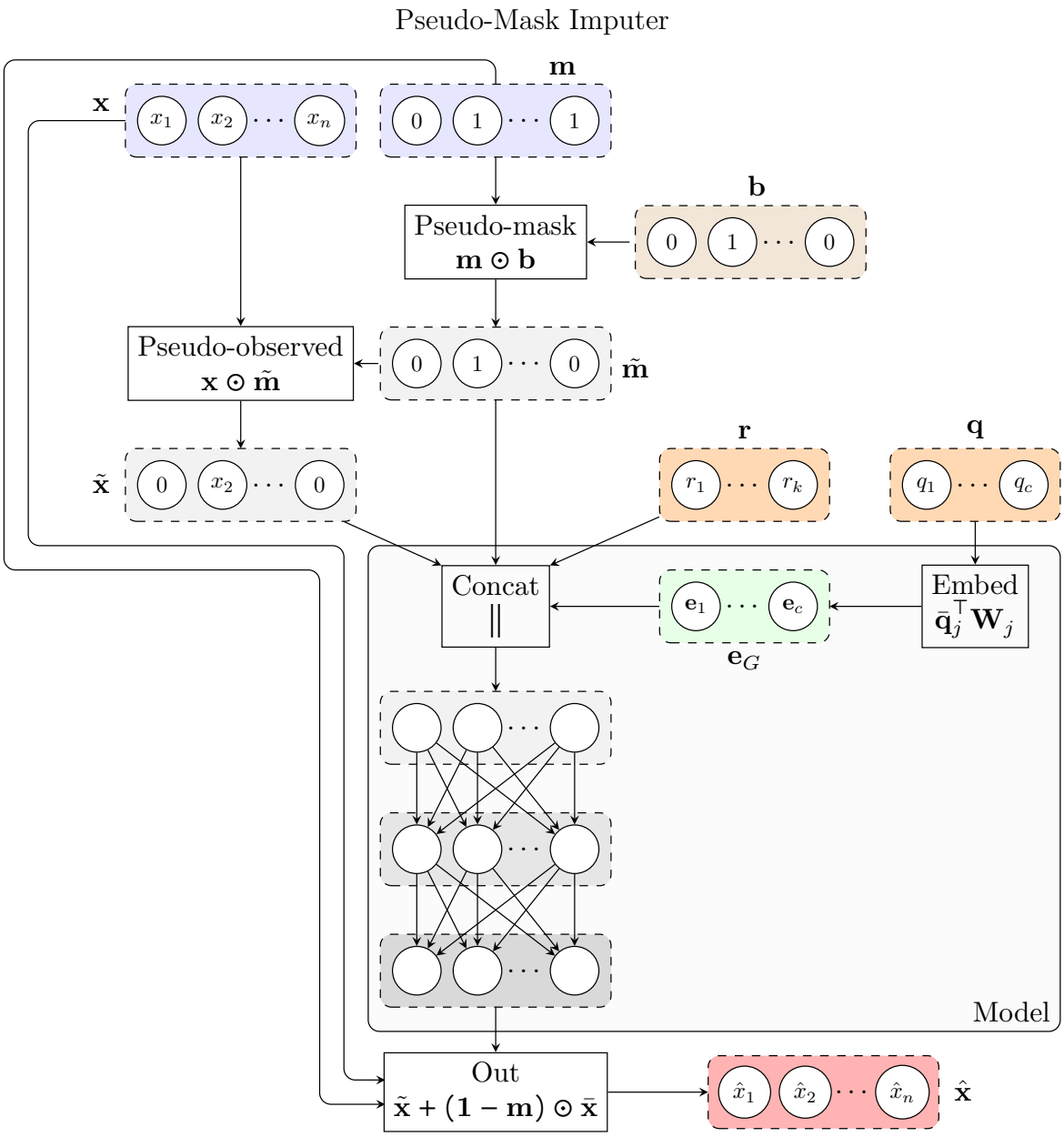


Fig. 4.1 Architecture of the Pseudo-Mask Imputer (PMI). The imputer receives gene expression values $\tilde{\mathbf{x}}$ with components missing according to a mask $\mathbf{m}$, and categorical (e.g., tissue type; $\mathbf{q}$) and numerical (e.g., age; $\mathbf{r}$) covariates, and outputs the imputed values $\bar{\mathbf{x}}$. The observed components of the imputer's output are then replaced by the observed values in $\tilde{\mathbf{x}}$, yielding the imputed sample $\hat{\mathbf{x}}$. The *pseudo-mask* mechanism masks out some of the observed components, which are then recovered at the output. Our architecture is flexible and supports inputs with different missing patterns.

4.1.3 Generative Adversarial Imputation Networks

The second method, which we call GAIN-GTEx, is based on Generative Adversarial Imputation Nets (GAIN; [184]). Generative Adversarial Networks have previously been used to synthesise transcriptomics *in-silico* [185, 14], but to our knowledge their applicability to gene expression imputation had not been studied prior to this work. Similar to generative adversarial networks (GANs; [186]), GAIN estimates a generative model via an adversarial process driven by the competition between two players, the *generator* and the *discriminator*.

Generator The generator aims at recovering missing data from partial gene expression observations, producing samples from the conditional $p(\mathbf{X}|\tilde{\mathbf{X}}, \mathbf{M}, \mathbf{R}, \mathbf{Q})$. Formally, we define the generator as a function $G : \mathbb{R}^n \times \mathbb{R}^n \times \{0,1\}^n \times \mathbb{R}^k \times \mathbb{N}^c \to \mathbb{R}^n$ that imputes expression values as follows:

$$\bar{\mathbf{x}} = G(\mathbf{x} \odot \mathbf{m}, \mathbf{z} \odot (\mathbf{1} - \mathbf{m}), \mathbf{m}, \mathbf{r}, \mathbf{q}),$$

where $\mathbf{z} \in \mathbb{R}^n$ is a vector sampled from a fixed noise distribution. Similar to GAIN, we mask the n-dimensional noise vector as $\mathbf{z} \odot (\mathbf{1} - \mathbf{m})$, encouraging a bijective association between noise components and genes. Before passing the output $\bar{\mathbf{x}}$ to the discriminator, we replace the prediction for the non-missing components by the original, observed expression values:

$$\hat{\mathbf{x}} = \mathbf{m} \odot \tilde{\mathbf{x}} + (\mathbf{1} - \mathbf{m}) \odot \bar{\mathbf{x}}$$

Discriminator The discriminator (also known as critic) takes the imputed samples $\hat{\mathbf{x}}$ and attempts to distinguish whether the expression value of each gene has been observed or produced by the generator. This is in contrast to the original GAN discriminator, which receives information from two input streams (generator and data distribution) and attempts to distinguish the true input source.

Formally, the discriminator is a function $D : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^k \times \mathbb{N}^c \to \mathbb{R}^n$ that outputs the probabilities $\hat{\mathbf{y}} \in \mathbb{R}^n$:

$$\hat{\mathbf{y}} = D(\hat{\mathbf{x}}, \mathbf{h}, \mathbf{r}, \mathbf{q}),$$

where the i-th component $\hat{y}_i$ is the probability of gene i being observed (as opposed to being imputed by the generator) for each $i \in \{1, \ldots, n\}$ and the vector $\mathbf{h} \in \mathbb{R}^n$ corresponds to the *hint* mechanism described in [184], which provides theoretical guarantees on the uniqueness of the global minimum for the estimation of $P(\mathbf{X}|\tilde{\mathbf{X}}, \mathbf{M}, \mathbf{R}, \mathbf{Q})$. Concretely, the role of the hint vector $\mathbf{h}$ is to *leak* some information about the mask $\mathbf{m}$

to the discriminator. Similar to GAIN, we define the hint $\mathbf{h}$ as follows:

$$\mathbf{h} = \mathbf{b} \odot \mathbf{m} + \frac{1}{2}(\mathbf{1} - \mathbf{b}) \qquad \mathbf{b} \sim B(1, p) \qquad p \sim U(\alpha, \beta), \tag{4.3}$$

where $\mathbf{b} \in \{0, 1\}^n$ is a binary vector that controls the amount of information from the mask $\mathbf{m}$ revealed to the discriminator. In contrast to GAIN, which discloses all but one components of the mask, we sample $\mathbf{b}$ from a Bernoulli distribution parametrised by a random probability $p \sim U(\alpha, \beta)$, where $\alpha \in [0, 1]$ and $\beta \in [\alpha, 1]$ are hyperparameters. This accounts for a high number of genes n and allows us to trade off the number of mask components that are revealed to the discriminator.

Optimisation Similarly to GAN and GAIN, we optimise the generator and discriminator adversarially, interleaving gradient updates for the discriminator and generator.

The discriminator aims at determining whether genes have been observed or imputed based on the imputed vector $\hat{\mathbf{x}}$, the covariates $\mathbf{q}$ and $\mathbf{r}$, and the hint vector $\mathbf{h}$. Since the hint vector $\mathbf{h}$ readily provides partial information about the ground truth $\mathbf{m}$ (Equation 4.3), we penalise D only for genes $i \in \{1, 2, ..., n\}$ such that $h_i = 0.5$, that is, genes whose corresponding mask value is unavailable to the discriminator. We achieve this via the following loss function $\mathcal{L}_D : \{0, 1\}^n \times \mathbb{R}^n \times \{0, 1\}^n \to \mathbb{R}$:

$$\mathcal{L}_D(\mathbf{m}, \hat{\mathbf{y}}, \mathbf{b}) = \frac{-1}{Z}(\mathbf{1} - \mathbf{b})^\top \Big(\mathbf{m} \odot \log \hat{\mathbf{y}} + (\mathbf{1} - \mathbf{m}) \odot (\mathbf{1} - \log \hat{\mathbf{y}})\Big),$$

where $Z = 1 + (\mathbf{1} - \mathbf{b})^\top (\mathbf{1} - \mathbf{b})$ is a normalisation term. The only difference with respect to the binary cross entropy loss function is the dot product involving $(\mathbf{1} - \mathbf{b})$, which we employ to ignore genes whose mask has been *leaked* to the discriminator through $\mathbf{h}$.

The generator aims at implicitly estimating $p(\mathbf{X}|\tilde{\mathbf{X}}, \mathbf{M}, \mathbf{R}, \mathbf{Q})$. Therefore, its role is not only to impute the expression corresponding to missing genes, but also to reconstruct the expression of the observed inputs. Similar to GAIN, in order to account for this and encourage a realistic imputation function, we use the following loss function $\mathcal{L}_G : \{0, 1\}^n \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \times \{0, 1\}^n \to \mathbb{R}$ for the generator:

$$\mathcal{L}_G(\mathbf{m}, \mathbf{x}, \bar{\mathbf{x}}, \hat{\mathbf{y}}, \mathbf{b}) = \frac{-1}{Z_1}\Big((\mathbf{1} - \mathbf{b}) \odot (\mathbf{1} - \mathbf{m})\Big)^\top \log \hat{\mathbf{y}} + \frac{\lambda}{Z_2}\mathbf{m}^\top (\mathbf{x} - \bar{\mathbf{x}})^2, \tag{4.4}$$

where $Z_1 = 1 + (\mathbf{1} - \mathbf{b})^\top (\mathbf{1} - \mathbf{b})$ and $Z_2 = \mathbf{m}^\top \mathbf{m}$ are normalisation terms, and $\lambda > 0$ is a hyperparameter. Intuitively, the first term in Equation 4.4 corresponds to the

adversarial loss, whereas the mean squared error (MSE) term accounts for the loss that the generator incurs in the reconstruction of the observed gene expression values.

Architecture We model the discriminator D and the generator G using neural networks. Similar to PMI, D and G leverage independent instances $\mathbf{e}^G$ and $\mathbf{e}^D$ of the categorical embeddings described in Equation 4.2. Specifically, we model the two players as follows:

$$G(\tilde{\mathbf{x}}, \tilde{\mathbf{z}}, \mathbf{m}, \mathbf{r}, \mathbf{q}) = \text{MLP}(\tilde{\mathbf{x}} \| \tilde{\mathbf{z}} \| \mathbf{m} \| \mathbf{r} \| \mathbf{e}^G) \qquad D(\hat{\mathbf{x}}, \mathbf{h}, \mathbf{r}, \mathbf{q}) = \text{MLP}(\hat{\mathbf{x}} \| \mathbf{h} \| \mathbf{r} \| \mathbf{e}^D),$$

where MLP denotes a multilayer perceptron and $\tilde{\mathbf{x}} = \mathbf{x} \odot \mathbf{m} \in \mathbb{R}^n$ and $\tilde{\mathbf{z}} = \mathbf{z} \odot (\mathbf{1} - \mathbf{m}) \in \mathbb{R}^n$ are the masked gene expression and noise input vectors, respectively. Figure 4.2 shows the architecture of both players.

Implementation For both PMI and GAIN-GTEx, we included the donor's age as numerical covariate in $\mathbf{r}$ and the tissue type, sex and cohort as categorical covariates in $\mathbf{q}$. We normalised the numerical variables via the standard score. For each categorical variable $q_j \in \{1, 2, ..., v_j\}$, we used the rule of thumb $d_j = \lfloor \sqrt{v_j} \rfloor + 1$ to set all the dimensions of the categorical embeddings. We used ReLU activations for each hidden layer in the MLP architectures of both PMI and GAIN (Equations 4.1.2 and 4.1.3).

We trained both models using the Adam optimiser [187]. We used batch normalisation [188] in the hidden layers of the MLPs, which yielded a significant speed-up to the training convergence according to our experiments. We used early stopping with a patience of 30. We present the rest of hyperparameters for each model, case study, and imputation scenario in Supplementary Information C.

4.1.4 Materials

Dataset The GTEx dataset is a public genomic resource of genetic effects on the transcriptome across a broad collection of human tissues, enabling linking of these regulatory mechanisms to trait and disease associations [12]. We downloaded the data from the GTEx portal and discarded underrepresented tissues (n=5), namely bladder, cervix (ectocervix, endocervix), fallopian tube, and kidney (medulla), yielding a dataset of 15,201 RNA-Seq samples collected from 49 tissues of 838 unique donors. We selected genes based on expression thresholds of ≥ 0.1 transcripts per kilobase million (TPM) in $\geq 20\%$ of samples and ≥ 6 reads (unnormalised) in $\geq 20\%$ of samples. We also selected the intersection of all the protein-coding genes among the 49 GTEx tissues tissues,

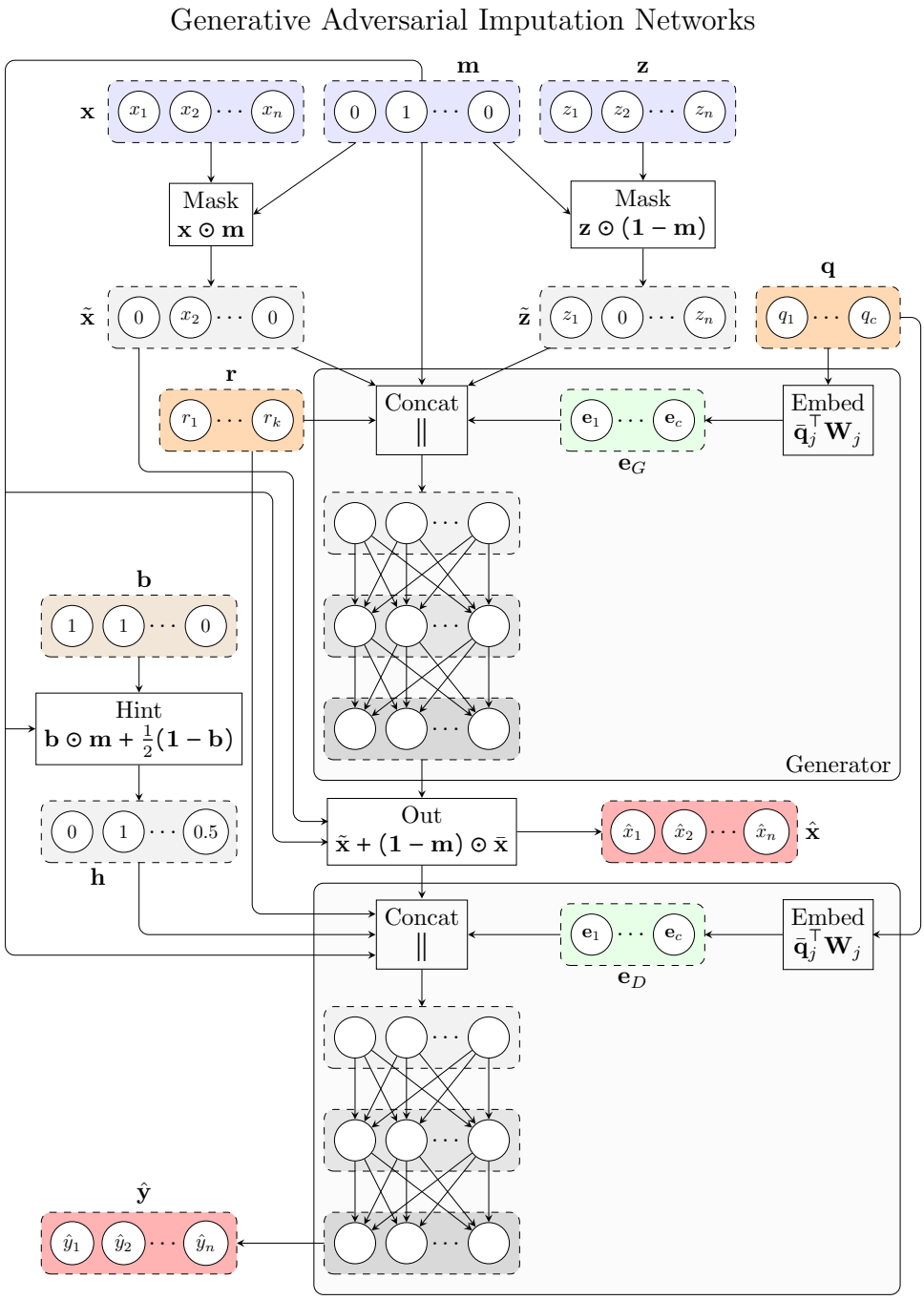


Fig. 4.2 Architecture of Generative Adversarial Imputation Networks for GTEx (GAIN-GTEx). The generator takes gene expression values $\tilde{\mathbf{x}}$ with missing components according to a mask $\mathbf{m}$, random noise $\mathbf{z}$, and categorical (e.g., tissue type; $\mathbf{q}$) and numerical (e.g., age; $\mathbf{r}$) covariates, and outputs the imputed values $\bar{\mathbf{x}}$. The observed components of the generator's output are then replaced by the actual observed expression values $\tilde{\mathbf{x}}$, yielding the imputed sample $\hat{\mathbf{x}}$. We simultaneously train a discriminator that receives $\hat{\mathbf{x}}$, the sample covariates $\mathbf{q}$ and $\mathbf{r}$, and the hint vector $\mathbf{h}$ — which provides partial information about the ground truth $\mathbf{m}$ — and produces the output $\hat{\mathbf{y}}$, whose i-th component $\hat{y}_i$ represents the probability of gene i being observed as opposed to being imputed by the generator.

yielding 12,557 unique genes. In addition to the expression data, we leveraged metadata about the sample donors, including sex, age, and cohort (post-mortem, surgical or organ donor).

Standardisation A large proportion of gene expression data in public repositories contains normalised values. Imputing the relative expression levels (in normalised data) vs absolute levels (in non-normalised data) is biologically meaningful and allows straightforward interpretation, with important applications, e.g., differential expression analysis (between disease individuals and controls) that is robust to expression outliers. To this end, following the standard GTEx processing pipeline for eQTL discovery (`https://github.com/broadinstitute/gtex-pipeline/tree/master/qtl`), we normalised the read counts across samples using the trimmed mean of M values (TMM) method [59] and applied an inverse normal transformation to the expression values for each gene. We further normalised the expression data via the standard score, so that the standardised expression values have mean 0 and standard deviation 1 for each gene across all samples.

Training, validation, and test splits To prevent any leakage of information between the training and test sets, we enforced all samples from the same donor to be within the same set. Concretely, we first flipped the GTEx donor identifiers (e.g., 111CU-1826 is flipped to 6281-UC111), we then sorted the reversed identifiers in alphabetical order, and we finally selected a suitable split point, forcing the two sets to be disjoint. After splitting the data, the training set, which we used to train the model, consisted of $\sim 60\%$ of the total samples. The validation set, which we used to optimise the method, consisted of $\sim 20\%$ of the total samples. The test set, on which we evaluated the final performance, contained the remaining $\sim 20\%$ of the data.

4.2 Results

We benchmarked all the baseline methods, including PMI and GAIN-GTEx, on two case studies and two imputation scenarios. In this section, we first present the benchmarking details. We then compare the performance of several imputation baselines on the two case studies and imputation scenarios. We finally study imputation generalisation of the proposed methods across missing rates and independent datasets.

4.2.1 Benchmarking details

Case studies To study the scalability of different imputation methods across the number of input variables, we considered the following case studies:

1. *Protein-coding genes.* As a first case study, we selected the intersection of all the protein-coding genes among the 49 GTEx tissues, resulting in a set of 12,557 unique genes. This case study is challenging for imputation methods that are not scalable across the number of input variables (i.e. genes).

2. *Genes in the Alzheimer's disease pathway.* We selected a subset of 273 genes from the Alzheimer's disease pathway extracted from the Kyoto Encyclopedia of Genes and Genomes (KEGG; [81]). This case study allows us to benchmark imputation methods that do not scale well with the number of variables.

Imputation scenarios We considered two realistic imputation scenarios:

1. *In-place imputation.* Our goal is to impute the missing values of a dataset $\mathcal{D} = \{(\mathbf{m} \odot \mathbf{x}, \mathbf{m}, \mathbf{r}, \mathbf{q})\}$ without access to the ground truth missing values $(\mathbf{1} - \mathbf{m}) \odot \mathbf{x}$. Importantly, for this scenario we assumed that the data is *missing completely at random* (MCAR; [189]), that is, the missingness does not depend on any of the observed nor unobserved variables.

2. *Inductive imputation.* Given a training dataset $\mathcal{D}_{train} = \{(\mathbf{x}, \mathbf{1}, \mathbf{r}, \mathbf{q})\}$ where all expression values $\mathbf{x} \in \mathbb{R}^n$ are observed, our goal is to impute the missing expression values of an independent test dataset $\mathcal{D}_{test} = \{(\tilde{\mathbf{x}}, \mathbf{m}, \mathbf{r}, \mathbf{q})\}$. Methods trained in inductive mode (e.g., on comprehensive datasets such as GTEx) can be used to perform imputation on small, independent datasets where the small number of samples is insufficient to train a model in in-place mode.

Baseline methods We compared PMI and GAIN-GTEx to several baseline methods:

- *Standard imputation methods.* We considered two simple gene expression imputation approaches: blood surrogate and median imputation. The use of blood, an easily accessible tissue, as a surrogate for difficult-to-acquire tissues is done in studies of biomarker discovery, diagnostics, and eQTLs, and in the development of model systems [190, 173]. For blood surrogate imputation, we imputed missing gene expression values in any given tissue with the corresponding values in whole blood for the same donor. For median imputation, we imputed missing values

with the median of the observed tissue-specific gene expression computed across donors.

- k-*Nearest Neighbours.* The k-Nearest Neighbours (k-NN) algorithm is an efficient method that is commonly used for imputation [191]. Here, we leveraged k-NN as a baseline for different values of k. This model estimates the missing values of a sample based on the values of the missing components in the k closest samples.

- *State-of-the-art methods.* We considered two state-of-the-art imputation methods: Multivariate Imputation by Chained Equations (MICE; [192]) and MissForest [193]. MICE leverages chained equations to create multiple imputations of missing data. The hyperparameters of MICE include the minimum/maximum possible imputed value for each component and the maximum number of imputation rounds. MissForest [193] is a non-parametric imputation method based on random forests trained on observed values to impute the missing values. Among others, the hyperparameters of MissForest include the number of trees in the forest and the number of features to consider when looking for the optimal split.

4.2.2 Imputation results

Table 4.1 Gene expression imputation performance with a missing rate of 50% across 3 runs (complete set of protein-coding genes). We did not report the R^2 scores for MICE and MissForest, because the runtime is longer than 7 days. GAIN-MSE-GTEx is a simplification of GAIN-GTEx optimised exclusively via the mean squared error term of the generator. Overall, GAIN-GTEx outperformed all the other methods in in-place imputation while PMI displayed the highest performance in inductive imputation.

	Scenario 1: In-place imputation		**Scenario 2: Inductive imputation**	
Method	R^2	Runtime (hours)	R^2	Runtime (hours)
MICE	–	–	–	–
MissForest	–	–	–	–
Blood surrogate	-0.693 ± 0.000	0.000 ± 0.000	-0.952 ± 0.000	0.000 ± 0.000
Median imputation	0.000 ± 0.000	0.001 ± 0.000	-0.009 ± 0.000	0.001 ± 0.000
1-NN imputation	0.179 ± 0.000	1.616 ± 0.004	0.203 ± 0.000	0.985 ± 0.003
5-NN imputation	0.461 ± 0.000	2.224 ± 0.107	0.482 ± 0.000	1.441 ± 0.096
10-NN imputation	0.468 ± 0.000	2.140 ± 0.035	0.495 ± 0.000	1.711 ± 0.160
GAIN-MSE-GTEx	0.637 ± 0.005	0.199 ± 0.074	0.638 ± 0.003	0.456 ± 0.053
GAIN-GTEx	$\mathbf{0.638 \pm 0.007}$	0.625 ± 0.294	0.636 ± 0.001	1.199 ± 0.157
PMI	0.479 ± 0.003	0.241 ± 0.024	$\mathbf{0.707 \pm 0.001}$	0.244 ± 0.019

Table 4.2 Gene expression imputation performance with a missing rate of 50% across 3 runs (for a subset of 273 genes from the Alzheimer's disease pathway).

	Scenario 1: In-place imputation		**Scenario 2: Inductive imputation**	
Method	R^2	Runtime (hours)	R^2	Runtime (hours)
MICE	$\mathbf{0.574 \pm 0.001}$	2.062 ± 0.335	0.569 ± 0.001	2.252 ± 0.096
MissForest (1 tree)	-0.147 ± 0.002	0.145 ± 0.002	-0.042 ± 0.003	0.575 ± 0.167
MissForest (10 trees)	0.458 ± 0.001	0.839 ± 0.176	0.514 ± 0.001	3.220 ± 0.371
MissForest (20 trees)	0.478 ± 0.000	1.836 ± 0.068	0.540 ± 0.000	4.842 ± 0.495
MissForest (100 trees)	0.493 ± 0.000	6.438 ± 0.498	0.561 ± 0.001	16.186 ± 1.709
Blood surrogate	-0.698 ± 0.002	0.000 ± 0.000	-0.971 ± 0.002	0.000 ± 0.000
Median imputation	0.001 ± 0.000	0.000 ± 0.000	-0.009 ± 0.000	0.000 ± 0.000
1-NN imputation	0.186 ± 0.001	0.037 ± 0.001	0.301 ± 0.000	0.021 ± 0.001
GAIN-MSE-GTEx	0.519 ± 0.001	0.038 ± 0.002	0.533 ± 0.001	0.045 ± 0.004
GAIN-GTEx	0.533 ± 0.001	0.139 ± 0.041	0.527 ± 0.003	0.569 ± 0.017
PMI	0.536 ± 0.001	0.048 ± 0.002	$\mathbf{0.630 \pm 0.011}$	0.037 ± 0.002

Method comparison We randomly masked out 50% of the values and studied the imputation performance of the baseline methods using two sets of genes: the complete set of protein-coding genes (Table 4.1) and genes from the Alzheimer's disease pathway (Table 4.2). We reported the per-gene coefficient of determination (R^2) between the predicted and ground-truth gene expression. This metric ranges from $-\infty$ to 1 and corresponds to the ratio of explained variance to the total variance. Negative scores indicate that the model predictions are worse than those of a baseline model that predicts the mean of the data. We averaged the results across 3 runs, each with different random masks.

Overall, GAIN-GTEx and PMI achieved comparable or superior imputation results compared to state-of-the-art imputation methods, i.e. MICE and MissForest, with substantially reduced runtime. In the case study of protein-coding genes, we halted the execution of MICE and MissForest after 7 days running on our server (CPU: Intel(R) Xeon(R) Processor E5-2630 v4. RAM: 125GB). The exceedingly long runtime of these methods highlights their poor scalability with the number of variables (i.e. genes), rendering them unfeasible in high-dimensional data regimes (e.g. gene expression datasets). In terms of imputation performance, GAIN-GTEx outperformed all the other methods ($R^2 = 0.638 \pm 0.007$) under the in-place imputation mode (Table 4.1; Scenario 1), while PMI showed the best overall performance ($R^2 = 0.707 \pm 0.001$) among all baseline methods (Table 4.1; Scenario 2) under the inductive imputation mode. In the case study involving genes from the Alzheimer's disease pathway, MICE attained the best imputation results ($R^2 = 0.574 \pm 0.001$) in the in-place imputation mode (Table 4.2; Scenario 1), followed by PMI ($R^2 = 0.536 \pm 0.001$) and GAIN-GTEx

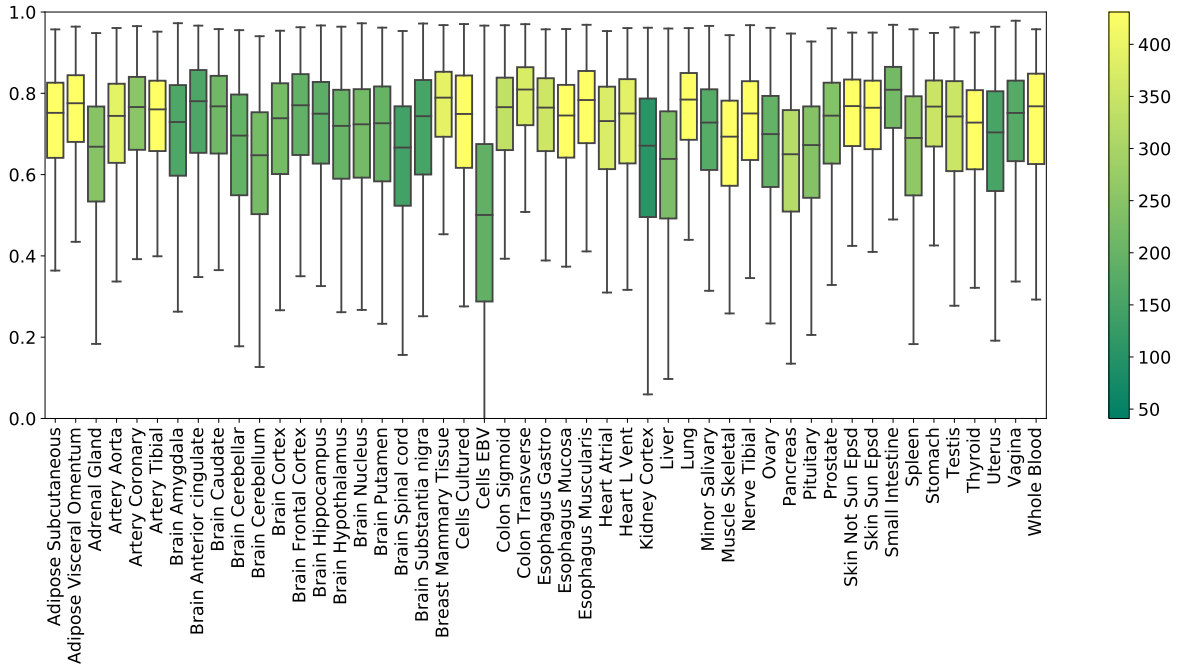


Fig. 4.3 R^2 imputation scores per GTEx tissue with a missing rate of 50% (PMI; inductive mode). Each box shows the distribution of the per-gene R^2 scores in the extended test set. The colour of each box represents the number of training samples of the corresponding tissue.

($R^2 = 0.533 \pm 0.001$). In inductive mode, PMI substantially outperformed all the other baselines ($R^2 = 0.630 \pm 0.011$) by a wide margin. In all case studies, we noted that GAIN-MSE-GTEx, a simplification of GAIN-GTEx optimised exclusively via the mean squared error term of the generator, performed reasonably well relative to GAIN-GTEx, suggesting that the mean squared error term of the loss function was driving the learning (Supplementary Information C.1).

Tissue-specific results We analysed the imputation performance across all 49 GTEx tissues (Figure 4.3). To obtain these results, we generated random masks with a missing rate of 50% for the test set, performed imputation using PMI, and plotted the distribution of 12,557 gene R^2 scores for each tissue. We observed that EBV-transformed lymphocytes, an accessible and renewable resource for functional genomics, were a notable outlier in imputation performance, consistent with studies about the transcriptional effect of EBV infection on the suitability of the cell lines as a model system for primary tissues [194]. Mean R^2 scores in the individual tissues ranged from ~ 0.5 (Epstein Barr virus-transformed lymphocytes; EBV) to ~ 0.78 (small intestine). Aside from the EBV-transformed lymphocytes, we noted that kidney

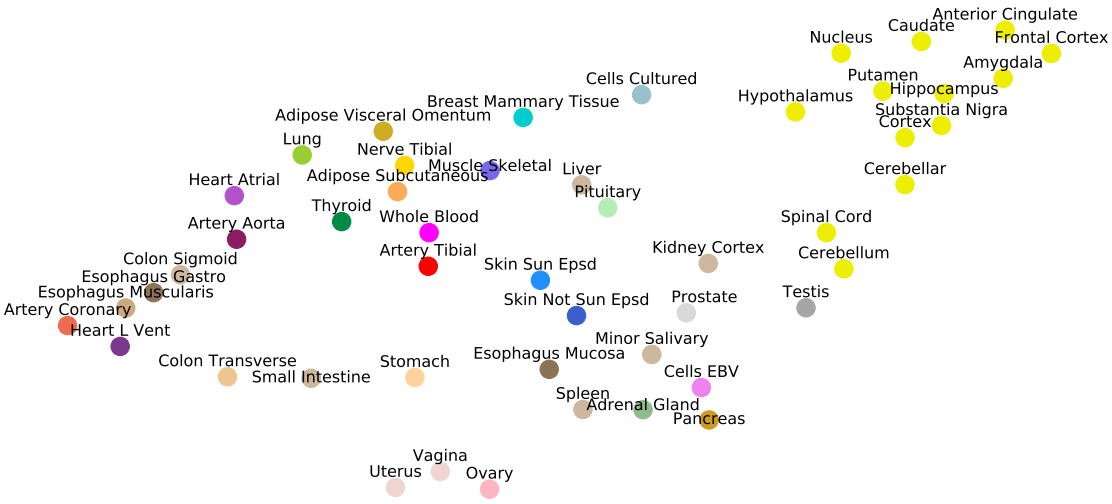


Fig. 4.4 UMAP visualisation of the tissue embeddings from the generator. Colours are assigned to conform to the GTEx Consortium conventions. Note that the central nervous system, consisting of the 13 brain regions, clusters together on the top right corner.

cortex, the tissue with the smallest sample size, had the highest variability in R^2 with an interquartile range of $Q_3 - Q_1 = 0.30$.

We then examined the tissue representations learnt by the models (Figure 4.4). Specifically, we plotted a UMAP representation [195] of the learnt tissue embeddings $\mathbf{W}_j \in \mathbb{R}^8$ from the generator of GAIN-GTEx (Equation 4.1), where j indexes the tissue dimension. Strikingly, the tissue representations showed strong clustering of biologically-related tissues, including the central nervous system (i.e., the 13 brain regions), the gastrointestinal system (e.g., the esophageal and colonic tissues), and the female reproductive tissues (i.e., uterus, vagina, and ovary). The clustering properties were robust across UMAP runs and could be similarly appreciated using other dimensionality reduction algorithms such as tSNE [196].

Cross-study results across missing rates To evaluate the cross-study relevance and generalisability of PMI and GAIN-GTEx, we leveraged the model trained on GTEx to perform imputation on The Cancer Genome Atlas (TCGA) gene expression data in acute myeloid leukemia (TCGA LAML; [197]), breast cancer (TCGA BRCA; [198]), and lung adenocarcinoma (TCGA LUAD; [199]). For each TCGA tissue and its *non-diseased* test counterpart on GTEx, we assessed imputation quality (Table 4.3) as well as the performance across varying missing rates (Figure 4.5). Prediction performance improved monotonically as we decreased the missing rate. Altogether,

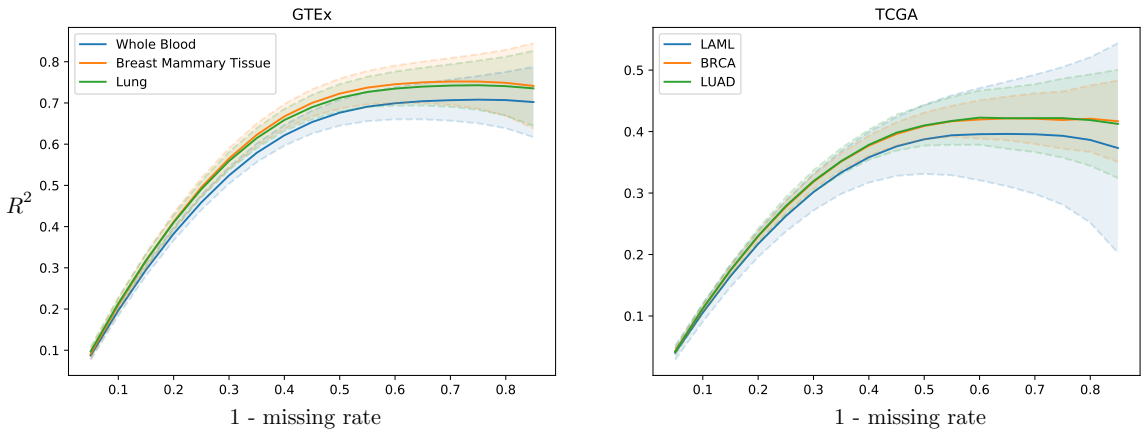


Fig. 4.5 GAIN-GTEx R^2 imputation scores per tissue across missing rate for 3 TCGA cancer types and their healthy counterpart in GTEx. The shaded area represents one standard deviation of the per-gene R^2 scores in the corresponding tissue. The greater the rate of missingness, the lower the performance.

PMI and GTEx-GAIN were robust to gene expression from multiple diseases in different tissues, with inferior yet stable performance on cancer expression data from TCGA, lending themselves to being used as tools to extend independent transcriptomic studies.

Table 4.3 Cross-study results for GAIN-GTEx and PMI trained on GTEx (inductive mode). We report the R^2 scores on data from 3 TCGA cancer types and their *healthy* counterpart on GTEx for a missing rate of 50%.

GAIN-GTEx		**PMI**	
Tissue	R^2	Tissue	R^2
TCGA LAML	0.386 ± 0.057	TCGA LAML	0.394 ± 0.065
TCGA BRCA	0.408 ± 0.023	TCGA BRCA	0.427 ± 0.023
TCGA LUAD	0.439 ± 0.034	TCGA LUAD	0.451 ± 0.050
GTEx Whole blood	0.678 ± 0.031	GTEx Whole blood	0.709 ± 0.034
GTEx Breast	0.724 ± 0.036	GTEx Breast	0.751 ± 0.039
GTEx Lung	0.713 ± 0.033	GTEx Lung	0.744 ± 0.035

Imputation results on genes from the Alzheimer's disease pathway We studied the per-gene imputation quality of 273 genes in the Alzheimer's disease pathway (Figure 4.6). Alzheimer's disease is characterised by the presence of amyloid plaques in the brain featuring amyloid-beta peptides, with various pathophysiological consequences on cellular processes. The pathway consists of genes that are involved in a number

of processes, including neuronal apoptosis, autophagy deficits, mitochondrial defect, and neurodegeneration. We observed that several genes in the pathway (e.g., PSMB6, COX6C, PSMD7, PSMA2, PSMD14, SDHB, TUBB1, TUBA8, FZD9, LPL, KIF5C, TUBB4A, TUBB2B, APOE) exhibited different distributions between brain and non-brain tissue types and the best highly imputed genes were enriched in known gene sets (Supplementary Information C.6).

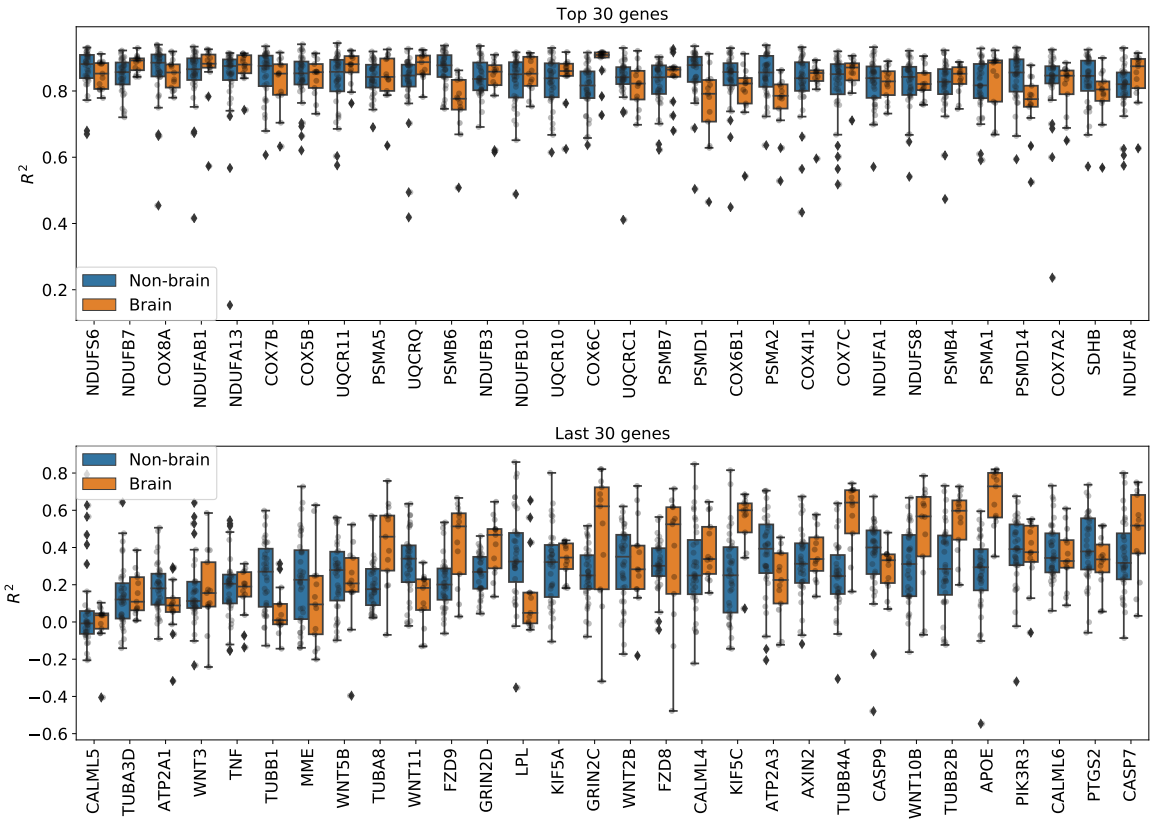


Fig. 4.6 Per-gene imputation R^2 scores on genes from the Alzheimer's disease pathway. Each point represents the average R^2 score in a tissue type. We note that some genes in the pathway (e.g., PSMB6, COX6C, PSMD7, PSMA2, PSMD14, SDHB, TUBB1, TUBA8, FZD9, LPL, KIF5C, TUBB4A, TUBB2B, APOE) exhibited different distributions between brain and non-brain tissue types.

4.3 Discussion

We developed two imputation approaches to gene expression, facilitating the reconstruction of a high-dimensional molecular trait that is central to disease biology and drug target discovery. The proposed methods, which we called Pseudo-Mask Imputer

(PMI) and GAIN-GTEx, were able to approximate the gene expression manifold from incomplete gene expression data and relevant covariates (potential global determinants of expression) and impute missing expression values. A characteristic feature of our architectures is the use of word embeddings, which enabled to learn distributed representations of the tissue types (Figure 4.4). Importantly, this allowed to condition the imputation algorithms on factors that drive gene expression, endowing the architectures with the ability to represent them in a biologically meaningful way.

We leveraged the most comprehensive human transcriptome resource available (GTEx), allowing us to test the performance of our method in a broad collection of tissues (Figure 4.3). The biospecimen repository includes commonly used surrogate tissues (whole blood and EBV transformed lymphocytes), central nervous system tissues from 13 brain regions, and a wide diversity of other primary tissues from *non-diseased* individuals. In particular, EBV-transformed lymphocytes, an accessible and renewable resource for functional genomics, were a notable outlier in imputation performance. This is perhaps not surprising, consistent with studies about the transcriptional effect of EBV infection on the suitability of the cell lines as a model system for primary tissues [194]. Interestingly, biologically similar tissues exhibited similar R^2 scores (Supplementary Information C.6).

The proposed approaches compared favourably to several existing imputation methods in terms of imputation performance and runtime (Table 4.1). We observed that standard approaches such as leveraging the expression of missing genes from a surrogate blood tissue yielded negative R^2 values and therefore did not perform well. Median imputation, although easy to implement, had limited predictive power. Imputation methods based on k-Nearest Neighbours were computationally feasible and yielded solid but poorer R^2 scores. In terms of state-of-the-art methods, MICE and MissForest were computationally prohibitive given the high dimensionality of the data and we halted their execution after running our experiments for 7 days. In particular, we performed an empirical study of the scalability of both methods (Supplementary Information C.2 and C.3) and observed that the runtime increases very rapidly with the number of genes. To alleviate this issue, we compared PMI and GAIN-GTEx with these methods on a subset of 273 genes from the Alzheimer's disease pathway (Table 4.2). Under the in-place imputation scenario (Alzheimer's disease pathway), MICE performed better than PMI, GAIN-GTEx, and MissForest (100 trees). Under the inductive imputation setting, PMI outperformed all the other methods by a large margin.

In terms of the comparison between PMI and GAIN-GTEx, our experiments suggest that the latter is generally harder to optimise (Supplementary Information C.1). In particular, GAIN resembles a deep autoencoder in that the supervised loss penalises the reconstruction error of the observed components. While this is a natural choice, autoencoder-like architectures are considerably sensitive to the user-definable bottleneck dimension. On one hand, a small number of units results in under-fitting. On the other hand, an excessively big bottleneck dimension allows the neural network to trivially *copy-paste* the observed components. In contrast, the loss function of PMI does not penalise the reconstruction error for the *pseudo-observed* components (e.g., the loss function of PMI penalises the prediction error of the *pseudo-missing* components, which are not provided as input at training time). Together with the fact that the *pseudo-mask* mechanism dynamically enlarges the training size, this subtlety allows training considerably bigger networks without over-fitting. Finally, we observed that a simplification of GAIN-GTEx, GAIN-MSE-GTEx, performed similarly well, suggesting that the mean squared error term of the generator's loss function is driving the learning process. In Supplementary Information C, we discuss our empirical findings about the adversarial loss of GAIN. For the purpose of reproducibility, as the gains of the adversarial loss appear to be small or negligible given our observations, we recommend training GAIN-GTEx without the adversarial term.

To evaluate the cross-study relevance of our method, we applied the trained models derived from GTEx (inductive mode) to perform imputation on The Cancer Genome Atlas gene expression data in acute myeloid leukemia, lung adenocarcinoma, and breast cancer. In addition to technical artifacts (e.g., batch effects), generalising to this data is highly challenging because the expression is largely driven by features of the disease such as chromosomal abnormalities, genomic instabilities, large-scale mutations, and epigenetic changes [200, 201]. Our results show that, despite these challenges, the methods were robust to gene expression from multiple diseases in different tissues (Table 4.3), lending themselves to being used as tools to extend independent transcriptomic studies. Next, we evaluated the imputation performance of PMI and GAIN-GTEx for a range of values for the missing rate (Figure 4.5 and Supplementary Information C). We noted that the performance was stable and that the greater the proportion of missing values, the lower the prediction performance. Finally, we analysed the imputation performance across genes from the Alzheimer's disease pathway (Figure 4.6) and across all genes (Supplementary Information C.6). The best-imputed genes were non-random and, indeed, clustered in some known pathways.

Broader Impact The study of the transcriptome is fundamental to our understanding of cellular and pathophysiological processes. High-dimensional gene expression data contain information relevant to a wide range of applications, including disease diagnosis [202], drug development [203], and evolutionary inference [177]. Thus, accurate and robust methods for imputation of gene expression have the enormous potential to enhance our molecular understanding of complex diseases, inform the search for novel drugs, and provide key insights into evolutionary processes. Here, we developed a methodology that attains state-of-the art performance in several scenarios in terms of imputation quality and execution time. Our analysis showed that the use of blood as a surrogate for difficult-to-acquire tissues, as commonly practised in biomedical research, may lead to substantially degraded performance, with important implications for biomarker discovery and therapeutic development. Our method generalises to gene expression in a disease class which has shown considerable health outcome disparities across population groups in terms of morbidity and mortality. Future algorithmic developments therefore hold promise for more effective detection, diagnosis, and treatment [204] and for improved implementation in clinical medicine [205]. Increased availability of transcriptomes in diverse human populations to enlarge our training data (a well-known and critical ethical challenge) should lead to further gains (i.e., decreased biases in results and reduced health disparities) [206]. This work has the potential to catalyse research into the application of deep learning to molecular reconstruction of cellular states and downstream gene mapping of complex traits [207, 171].

Conclusion In this work, we developed two methods for gene expression imputation, which we named PMI and GAIN-GTEx. To increase the applicability of the proposed methods, we trained them on RNA-Seq data from the Genotype-Tissue Expression project, a reference resource that has generated a comprehensive collection of transcriptomes in a diverse set of tissues. A characteristic feature of our architectures is the use of word embeddings to learn distributed representations for the tissue types. Our approaches compared favourably to several standard and state-of-the-art imputation methods in terms of predictive performance and runtime, and generalised to transcriptomics data from 3 cancer types of the The Cancer Genome Atlas. PMI and GAIN-GTEx show optimal performance among the methods in inductive and in-place imputation, respectively, on the protein-coding genes. This work can facilitate the straightforward integration and cost-effective repurposing of large-scale RNA biorepos-

itories into genomic studies of disease, with high applicability across diverse tissue types.

Chapter 5

Multi-tissue imputation of gene expression

Sequencing technologies have enabled profiling the transcriptome at tissue and single-cell resolutions, with great potential to unveil intra- and multi-tissue molecular phenomena such as cell signalling and disease mechanisms. Due to the invasiveness of the sampling process, gene expression is usually measured independently in easy-to-acquire tissues, leading to an incomplete picture of an individual's physiological state and necessitating effective multi-tissue integration methodologies.

A question of fundamental biological significance is to what extent the transcriptomes of difficult-to-acquire tissues and cell types can be inferred from those of accessible ones [21, 12]. Due to their ease of collection, accessible tissues such as whole blood could have great utility for diagnosis and monitoring of pathophysiological conditions through metabolites, signalling molecules, and other biomarkers, including possible transcriptome-level associations [208]. Moreover, all human somatic cells share the same genetic information, which may regulate expression in a context-dependent and temporal manner, partially explaining tissue- and cell-type-specific gene expression variation. Computational models that exploit these patterns could therefore be used to impute the transcriptomes of uncollected cell types and tissues, with potential to elucidate the biological mechanisms regulating a diverse range of developmental and physiological processes.

Multi-tissue imputation is a central problem in transcriptomics with broad implications for fundamental biological research and translational science. The methodological

---

The research presented in this chapter has been conducted in collaboration with Chaitanya K. Joshi, Dobrik Georgiev, Phillip Lin, Bianca Dumitrascu, Eric R. Gamazon, and Pietro Liò

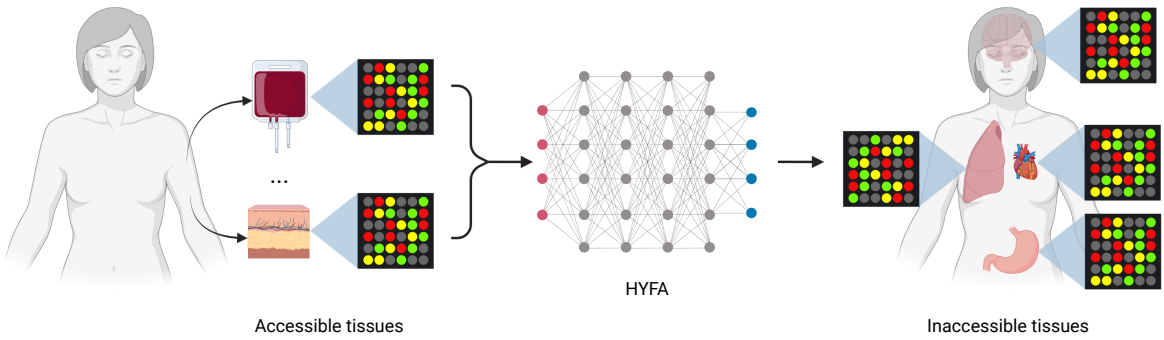


Fig. 5.1 Overview of the multi-tissue gene expression imputation problem. HYFA processes gene expression from a variable number of collected tissues (e.g. accessible tissues) and infers the transcriptomes of uncollected tissues.

problem can powerfully influence downstream applications, including performing differential expression analysis, identifying regulatory mechanisms, determining co-expression networks, and enabling drug target discovery. In practice, in experimental follow-up or clinical application, the task includes the special case of determining a good proxy or easily-assayed system for *causal* tissues and cell types. Multi-tissue integration methods can also be applied to harmonise large collections of RNA-seq datasets from diverse institutions, consortia, and studies [209] — each potentially affected by technical artefacts — and to characterise gene expression co-regulation across tissues. Reconstruction of unmeasured gene expression across a broad collection of tissues and cell types from available reference transcriptome panels may expand our understanding of the molecular origins of complex traits and of their context specificity.

Several methods have traditionally been employed to impute uncollected gene expression. Leveraging a surrogate tissue has been widely used in studies of biomarker discovery, diagnostics, and expression Quantitative Trait Loci (eQTLs), and in the development of model systems [210–212, 173, 190]. Nonetheless, gene expression is known to be tissue and cell-type specific, limiting the utility of a proxy tissue. Other related studies impute tissue-specific gene expression from genetic information [207]. Wang et al. [213] propose a mixed-effects model to infer uncollected data in multiple tissues from eQTLs. Sul et al. [214] introduce a model termed Meta-Tissue that aggregates information from multiple tissues to increase statistical power of eQTL detection. However, these approaches do not model the complex non-linear relationships between measured and unmeasured gene expression traits among tissues and cell types, and individual-level genetic information (such as at eQTLs) is subject to privacy concerns and often unavailable.

Computationally, multi-tissue transcriptome imputation is challenging because the data dimensionality scales rapidly with the number of genes and tissues, often leading to overparameterised models. TEEBoT [21] addresses this issue by employing principal component analysis (PCA) — a non-parametric dimensionality reduction method — to project the data into a low-dimensional manifold, followed by linear regression to predict target gene expression from the principal components. However, this technique does not account for non-linear effects and can only handle a single reference tissue, i.e. whole blood. Approaches such as standard multilayer perceptrons can exploit non-linear patterns, but are massively overparameterised and computationally infeasible.

To address these challenges, we present HYFA (**Hy**pergraph **Fa**ctorisation), a parameter-efficient graph representation learning approach for joint multi-tissue and cell-type gene expression imputation. HYFA represents multi-tissue gene expression in a hypergraph of individuals, metagenes, and tissues, and learns factorised representations via a specialised message passing neural network operating on the hypergraph. In contrast to existing methods, HYFA supports a variable number of reference tissues, increasing the statistical power over single-tissue approaches, and incorporates inductive biases to exploit the shared regulatory architecture of tissues and genes. In performance comparison, HYFA attains improved performance over TEEBoT and standard imputation methods across a broad range of tissues from the Genotype-Tissue Expression (GTEx) project (v8) [12]. Through transfer learning on a paired single-nucleus RNA-seq dataset (GTEx-v9) [215], we further demonstrate the ability of HYFA to resolve cell-type signatures — average gene expression across cells for a given cell-type, tissue, and individual — from bulk gene expression. Thus, HYFA may provide a unifying transcriptomic methodology for multi-tissue imputation and cell-type deconvolution. In post-imputation analysis, application of eQTL mapping on the fully-imputed GTEx data yields a substantial increase in number of detected replicable eQTLs. HYFA is publicly available at `https://github.com/rvinas/HYFA`.

5.1 Methodology

5.1.1 Problem formulation

Suppose we have a transcriptomics dataset of N individuals/donors, T tissues, and G genes. For each individual $i \in \{1, ..., N\}$, let $\boldsymbol{X}_i \in \mathbb{R}^{T \times G}$ be the gene expression values in T tissues and define the donor's demographic information by $\boldsymbol{u}_i \in \mathbb{R}^C$, where C is the number of covariates. Denote by $\boldsymbol{x}_i^{(k)}$ the k-th entry of $\boldsymbol{X}_i$, corresponding to

the expression values of donor i measured in tissue k. For a given donor i, let $\mathcal{T}(i)$ represent the collection of tissues with measured expression values. These sets might vary across individuals. Let $\tilde{\boldsymbol{X}}_i \in (\mathbb{R} \cup \{*\})^{T \times G}$ be the measured gene expression values, where $*$ denotes unobserved, so that $\tilde{\boldsymbol{x}}_i^{(k)} = \boldsymbol{x}_i^{(k)}$ if $k \in \mathcal{T}(i)$ and $\tilde{\boldsymbol{x}}_i^{(k)} = *$ otherwise. Our goal is to infer the uncollected values in $\tilde{\boldsymbol{X}}_i$ by modelling the distribution $p(\mathbf{X} = \boldsymbol{X}_i | \tilde{\mathbf{X}} = \tilde{\boldsymbol{X}}_i, \mathbf{U} = \boldsymbol{u}_i)$.

5.1.2 Multi-tissue model

An important challenge of modelling multi-tissue gene expression is that a different set of tissues might be collected for each individual. Moreover, the data dimensionality scales rapidly with the total number of tissues and genes. To address these problems, we represent the data in a hypergraph and develop a parameter-efficient neural network that operates on this hypergraph. Throughout, we make use of the concept of *metagenes* [216, 217]. Each *metagene* characterises certain gene expression patterns and is defined as a positive linear combination of multiple genes [216, 217].

Hypergraph representation

We represent the data in a hypergraph consisting of three types of nodes: donor, tissue, and *metagene* nodes.

Mathematically, we define a hypergraph $\mathcal{G} = \{\mathcal{V}_d \cup \mathcal{V}_m \cup \mathcal{V}_t, \mathcal{E}\}$, where $\mathcal{V}_d$ is a set of donor nodes, $\mathcal{V}_m$ is a set of *metagene* nodes, $\mathcal{V}_t$ is a set of tissue nodes, and $\mathcal{E}$ is a set of multi-attributed hyperedges. Each hyperedge connects an individual i with a *metagene* j and a tissue k if $k \in \mathcal{T}(i)$, where $\mathcal{T}(i)$ are the collected tissues of individual i. The set of all hyperedges is defined as $\mathcal{E} = \{(i, j, k, \boldsymbol{e}_{ij}^{(k)}) \mid (i, j, k) \in \mathcal{V}_d \times \mathcal{V}_m \times \mathcal{V}_t, k \in \mathcal{T}(i)\}$, where $\boldsymbol{e}_{ij}^{(k)}$ are hyperedge attributes that describe characteristics of the interacting nodes, i.e. features of *metagene* j in tissue k for individual i.

The hypergraph representation allows representing data in a flexible way, generalising the bipartite graph representation from [218]. On the one hand, using a single *metagene* results in a bipartite graph where each edge connects an individual i with a tissue k. In this case, the edge attributes $\boldsymbol{e}_{i1}^{(k)}$ are derived from the gene expression $\boldsymbol{x}_i^{(k)}$ of individual i in tissue k. On the other hand, using multiple *metagenes* leads to a hypergraph where each individual i is connected to tissue k through multiple hyperedges. For example, it is possible to construct a hypergraph where genes and *metagenes* are related by a one-to-one correspondence, with hyperedge attributes $\boldsymbol{e}_{ij}^{(k)}$ derived directly from expression $x_{ij}^{(k)}$. The number of *metagenes* thus controls a

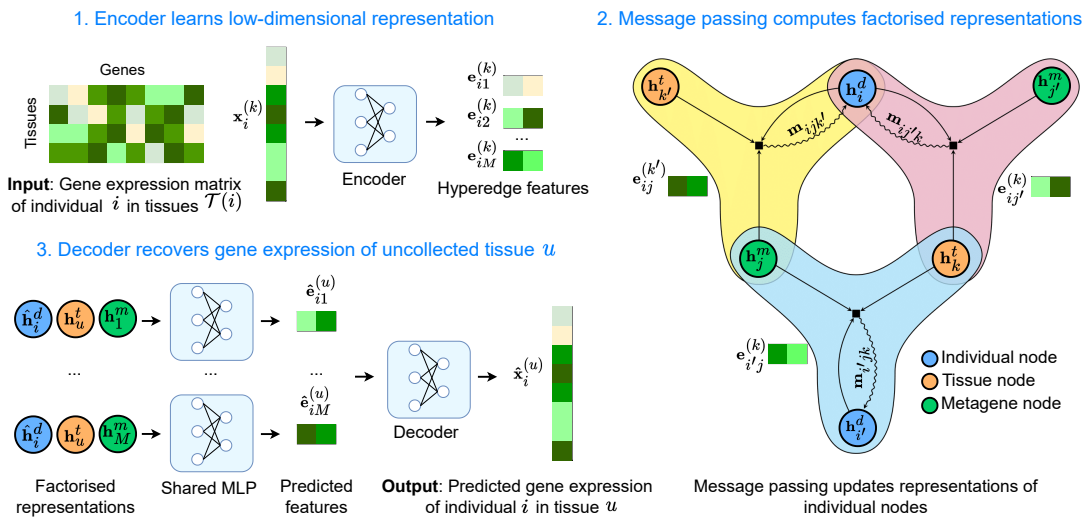


Fig. 5.2 Workflow of HYFA. The model receives as input a variable number of gene expression samples $\boldsymbol{x}_i^{(k)}$ corresponding to the collected tissues $k \in \mathcal{T}(i)$ of a given individual i. The samples $\boldsymbol{x}_i^{(k)}$ are fed through an encoder that computes low-dimensional representations $\boldsymbol{e}_{ij}^{(k)}$ for each metagene $j \in 1..M$. A *metagene* is a latent, low-dimensional representation that captures certain gene expression patterns of the high-dimensional input sample. These representations are then used as hyperedge features in a message passing neural network that operates on a hypergraph. In the hypergraph representation, each hyperedge labelled with $\boldsymbol{e}_{ij}^{(k)}$ connects an individual i with metagene j and tissue k if tissue k was collected for individual i, i.e. $k \in \mathcal{T}(i)$. Through message passing, HYFA learns factorised representations of individual, tissue, and metagene nodes. To infer the gene expression of an uncollected tissue u of individual i, the corresponding factorised representations are fed through a multilayer perceptron (MLP) that predicts low-dimensional features $\boldsymbol{e}_{ij}^{(u)}$ for each metagene $j \in 1..M$. HYFA finally processes these latent representations through a decoder that recovers the uncollected gene expression sample $\hat{\boldsymbol{x}}_{ij}^{(u)}$.

spectrum of hypergraph representations and, as we shall see, can help alleviate the inherent over-squashing problem of graph neural networks.

Message passing neural network

Given the hypergraph representation of the multi-tissue transcriptomics dataset, we now present a parameter-efficient graph neural network to learn donor, metagene, and tissue embeddings, and infer the expression values of the unmeasured tissues. We start by computing hyperedge attributes from the multi-tissue expression data. Then, we initialise the embeddings of all nodes in the hypergraph, construct the message

passing neural network, and define an inference model that builds on the latent node representations obtained via message passing.

Computing hyperedge attributes We first reduce the dimensionality of the measured transcriptomics values. For every individual i and measured tissue k, we project the corresponding gene expression values $\boldsymbol{x}_i^{(k)}$ into low-dimensional *metagene* representations $\boldsymbol{e}_{ij}^{(k)}$:

$$\boldsymbol{e}_{ij}^{(k)} = \mathrm{ReLU}(\boldsymbol{W}_j \boldsymbol{x}_i^{(k)}) \qquad \forall j \in 1..M, \tag{5.1}$$

where M, the number of metagenes, is a user-definable hyperparameter and $\boldsymbol{W}_j \forall j \in 1..M$ are learnable parameters. In addition to characterising groups of functionally similar genes, employing metagenes reduces the number of messages being aggregated for each node, addressing the over-squashing problem of graph neural networks (Supplementary Information D.2).

Initial node embeddings We initialise the node features of the individual $\mathcal{V}_p$, metagene $\mathcal{V}_m$, and tissue $\mathcal{V}_t$ partitions with learnable parameters and available information. For metagene and tissue nodes, we use learnable embeddings as initial node values. The idea is that these weights, which will be approximated through gradient descent, should summarise relevant properties of each metagene and tissue. We initialise the node features of each individual with the available demographic information $\boldsymbol{u}_i$ of each individual i (we use age and sex). We encode sex as a binary value and age as a float normalised by 100 (e.g. age 30 is encoded as 0.30). Importantly, this formulation allows transfer learning between sets of distinct donors.

Message passing layer We develop a custom graph neural network (GNN) layer to compute latent donor embeddings by passing messages along the hypergraph. At each layer of the GNN, we perform message passing to iteratively refine the individual node embeddings. We do not update the tissue and metagene embeddings during message passing, in a similar vein as knowledge graph embeddings [219], because their node embeddings already consist of learnable weights that are updated through gradient descent. Sending messages to these nodes would also introduce a dependency between individual nodes and tissue and metagene features (and, by transitivity, dependencies between individuals). However, if we foresee that unseen entities will be present at test time (e.g. new tissue types), our approach can be extended by initialising their

node features with constant values and introducing node-type-specific message passing equations.

Mathematically, let $\{\boldsymbol{h}_1^d, ..., \boldsymbol{h}_N^d\}$, $\{\boldsymbol{h}_1^m, .., \boldsymbol{h}_M^m\}$, and $\{\boldsymbol{h}_1^t, .., \boldsymbol{h}_T^t\}$ be the donor, metagene, and tissue node embeddings, respectively. At each layer of the GNN, we compute refined individual embeddings $\{\hat{\boldsymbol{h}}_1^d, ..., \hat{\boldsymbol{h}}_N^d\}$ as follows:

$$\begin{aligned} \hat{\boldsymbol{h}}_i^d &= \phi_h(\boldsymbol{h}_i^d, \boldsymbol{m}_i) \\ \boldsymbol{m}_i &= \sum_{j=1}^{M} \sum_{k \in \mathcal{T}(i)} \phi_a(\boldsymbol{h}_j^m, \boldsymbol{h}_k^t, \boldsymbol{m}_{ijk}) \\ \boldsymbol{m}_{ijk} &= \phi_e(\boldsymbol{h}_i^d, \boldsymbol{h}_j^m, \boldsymbol{h}_k^t, \boldsymbol{e}_{ij}^{(k)}), \end{aligned} \tag{5.2}$$

where the functions ϕ_e and ϕ_h are edge and node operations that we model as multilayer perceptrons (MLP), and ϕ_a is a function that determines the aggregation behavior. In its simplest form, choosing $\phi_a(\boldsymbol{h}_j^m, \boldsymbol{h}_k^t, \boldsymbol{m}_{ijk}) = \frac{1}{M|\mathcal{T}(i)|}\boldsymbol{m}_{ijk}$ results in average aggregation. We analyse the time complexity of the message passing layer in Supplementary Information D.1. Optionally, we can stack several message passing layers to increase the expressivity of the model.

The architecture is flexible and may be extended as follows:

- Incorporation of information about the individual embeddings $\boldsymbol{h}_i^d$ into the aggregation mechanism ϕ_a.
- Incorporation of target tissue embeddings $\boldsymbol{h}_u^t$, for a given target tissue u, into the aggregation mechanism ϕ_a.
- Update hyperedge attributes $\boldsymbol{e}_{ij}^{(k)}$ at every layer.

Aggregation mechanism In practice, the proposed hypergraph neural network suffers from a bottleneck. In the aggregation step, the number of messages being aggregated is $M|\mathcal{T}(i)|$ for each individual i. In the worst case, when all genes are used as metagenes (i.e. $M = G$; it is estimated that humans have around $G \approx 25,000$ protein-coding genes), this leads to serious over-squashing — large amounts of information are compressed into fixed-length vectors [220]. Fortunately, choosing a small number of metagenes reduces the dimensionality of the original transcriptomics values which in turn alleviates the over-squashing and scalability problems (Supplementary Information B). To further attenuate over-squashing, we propose an attention-based aggregation

mechanism ϕ_a that weighs metagenes according to their relevance in each tissue:

$$\begin{aligned}\phi_a(\boldsymbol{h}_j^m, \boldsymbol{h}_k^t, \boldsymbol{m}_{ijk}) &= \alpha_{jk}\boldsymbol{m}_{ijk} \\ \alpha_{jk} &= \frac{\exp e(\boldsymbol{h}_j^m, \boldsymbol{h}_k^t)}{\sum_v \exp e(\boldsymbol{h}_v^m, \boldsymbol{h}_k^t)} \\ e(\boldsymbol{h}_j^m, \boldsymbol{h}_k^t) &= \mathbf{a}^\top \mathrm{LeakyReLU}\Big(\mathbf{W}[\boldsymbol{h}_j^m || \boldsymbol{h}_k^t]\Big),\end{aligned}$$

where $||$ is the concatenation operation and $\mathbf{a}$ and $\mathbf{W}$ are learnable parameters. The proposed attention mechanism, which closely follows the neighbour aggregation method of graph attention networks [221, 222], computes dynamic weighting coefficients that prioritise messages coming from important metagenes. Optionally, we can leverage multiple heads [223] to learn multiple modes of interaction and increase the expressivity of the model.

Hypergraph model The hypergraph model, which we define as f, computes latent individual embeddings $\hat{\boldsymbol{h}}_i^d$ from incomplete multi-tissue expression profiles as $\hat{\boldsymbol{h}}_i^d = f(\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i)$.

5.1.3 Downstream imputation tasks

The resulting donor representations $\hat{\boldsymbol{h}}_i^d$ summarise information about a variable number of tissue types collected for donor i, in addition to demographic information. We leverage these embeddings for two downstream tasks: inference of gene expression in uncollected tissues and prediction of cell-type signatures.

Inference of gene expression in uncollected tissues

Predicting the transcriptomic measurements $\hat{\boldsymbol{x}}_i^{(k)}$ of a tissue k (e.g. uncollected) is achieved by first recovering the latent metagene values $\hat{\boldsymbol{e}}_{ij}^{(k)}$ for all metagenes $j \in 1..M$, a hyperedge-level prediction task, and then decoding the gene expression values from the predicted metagene representations $\hat{\boldsymbol{e}}_{ij}^{(k)}$ with an appropriate probabilistic model.

Prediction of hyperedge attributes To predict the latent metagene attributes $\hat{\boldsymbol{e}}_{ij}^{(k)}$ for all $j \in 1..M$, we employ a multilayer perceptron that operates on the *factorised* metagene $\boldsymbol{h}_j^m$ and tissue representations $\boldsymbol{h}_k^t$ as well as the latent variables $\hat{\boldsymbol{h}}_i^d$ of individual i:

$$\hat{\boldsymbol{e}}_{ij}^{(k)} = \mathrm{MLP}(\hat{\boldsymbol{h}}_i^d, \boldsymbol{h}_j^m, \boldsymbol{h}_k^t),$$

where the MLP is shared for all combinations of metagenes, individuals, and tissues.

Negative binomial imputation model For raw count data, we use a negative binomial likelihood. To decode the gene expression values for a tissue k of individual i, we define the following probabilistic model $p(\boldsymbol{x}_i^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k)$:

$$
\begin{aligned}
p(\boldsymbol{x}_i^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k) &= \prod_j^G p(x_{ij}^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k) \\
p(x_{ij}^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k) &= \mathrm{NB}(x_{ij}^{(k)}; \mu_{ij}^{(k)}, \theta_{ij}^{(k)}),
\end{aligned}
$$

where NB is a negative binomial distribution. The mean $\mu_{ij}^{(k)}$ and dispersion $\theta_{ij}^{(k)}$ parameters of this distribution are computed as follows:

$$
\begin{aligned}
\boldsymbol{\mu}_i^{(k)} &= l_i^{(k)} \boldsymbol{s}_i^{(k)} \\
\boldsymbol{s}_i^{(k)} &= \mathrm{softmax}\Big(\boldsymbol{W}_s \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_s\Big) \\
\boldsymbol{\theta}_i^{(k)} &= \exp\Big(\boldsymbol{W}_\theta \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\theta\Big) \\
\hat{\boldsymbol{e}}_i^{(k)} &= \mathrm{MLP}\Big(\Big\|_{j=1}^M \hat{\boldsymbol{e}}_{ij}^{(k)}\Big),
\end{aligned}
$$

where $\mathbf{s}_i^{(k)}$ are mean gene-wise proportions; $\boldsymbol{W}_s$, $\boldsymbol{W}_\theta$, $\boldsymbol{b}_s$, and $\boldsymbol{b}_\theta$ are learnable parameters; and $l_i^{(k)}$ is the library size, which is modelled with a log-normal distribution

$$
\begin{aligned}
\log l_i^{(k)} &\sim \mathcal{N}(l_i^{(k)}; \nu_i^{(k)}, \omega_i^{(k)}) \\
\nu_i^{(k)} &= \boldsymbol{W}_\nu \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\nu \\
\omega_i^{(k)} &= \exp\Big(\boldsymbol{W}_\omega \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\omega\Big),
\end{aligned}
$$

where $\boldsymbol{W}_\nu$, $\boldsymbol{W}_\omega$, $\boldsymbol{b}_\nu$, and $\boldsymbol{b}_\omega$ are learnable parameters. Optionally, we can use the observed library size.

Gaussian imputation model For normalised gene expression data (i.e. inverse normal transformed data), we use the following Gaussian likelihood:

$$
\begin{aligned}
p(\boldsymbol{x}_i^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k) &= \prod_j^G p(x_{ij}^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k) \\
p(x_{ij}^{(k)}|\hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k) &= \mathcal{N}(x_{ij}^{(k)}; \mu_{ij}^{(k)}, \sigma^{2(k)}_{ij}),
\end{aligned}
$$

where the mean $\mu_{ij}^{(k)}$ and standard deviation $\sigma_{ij}^{(k)}$ are computed as follows:

$$\begin{aligned}\boldsymbol{\mu}_i^{(k)} &= \boldsymbol{W}_\mu \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\mu \\ \boldsymbol{\sigma}_i^{(k)} &= \text{softplus}\Big(\boldsymbol{W}_\sigma \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\sigma\Big) \\ \hat{\boldsymbol{e}}_i^{(k)} &= \text{MLP}\Big(\Big\|_{j=1}^{M} \hat{\boldsymbol{e}}_{ij}^{(k)}\Big),\end{aligned}$$

where $\boldsymbol{W}_\mu$, $\boldsymbol{W}_\sigma$, $\boldsymbol{b}_\mu$, and $\boldsymbol{b}_\sigma$ are learnable parameters and $\text{softplus}(x) = \log\Big(1+\exp(x)\Big)$.

Optimisation We optimise the model to maximise the imputation performance on a dynamic subset of observed tissues, that is, tissues that are masked out at train time, similar to Chapter 4 [2]. For each individual i, we randomly select a subset $\mathcal{C} \subset \mathcal{T}(i)$ of *pseudo-observed* tissues and treat the remaining tissues $\mathcal{U} = \mathcal{T}(i) - \mathcal{C}$ as unobserved (*pseudo-missing*). We then compute the individual embeddings $\hat{\boldsymbol{h}}_i^d$ using the gene expression of *pseudo-observed* tissues $\mathcal{C}$ and minimise the loss:

$$\mathcal{L}(\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i, \mathcal{C}, \mathcal{U}) = -\frac{1}{|\mathcal{U}|} \sum_{k \in \mathcal{U}} \log p(\boldsymbol{x}_i^{(k)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k),$$

which corresponds to the average negative log-likelihood across *pseudo-missing* tissues. Importantly, the *pseudo-mask* mechanism generates different sets of *pseudo-missing* tissues for each individual, effectively enlarging the number of training examples and regularising our model. We report the hyperparameters in Supplementary Information D.2 and summarise the training algorithm in Supplementary Information D.4. HYFA can also be optimised via variational inference (Supplementary Information D.5).

Inference of gene expression from uncollected tissues At test time, we infer the gene expression values $\hat{\boldsymbol{x}}_i^{(v)}$ of an uncollected tissue v from a given donor i via the mean, i.e. $\hat{\boldsymbol{x}}_i^{(v)} = \boldsymbol{\mu}_i^{(v)}$. Alternatively, we can draw random samples from the conditional predictive distribution $p(\boldsymbol{x}_i^{(k)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k)$.

Prediction of cell-type signatures

We next consider the problem of imputing cell-type signatures in a tissue of interest. We define a cell-type signature as the sum of gene expression profiles across cells of a given cell-type in a certain tissue. Formally, let $\boldsymbol{x}_i^{(k,q)}$ be the gene expression signature of cell-type q in a tissue of interest k of individual i. Our goal is to infer $\boldsymbol{x}_i^{(k,q)}$ from the multi-tissue gene expression measurements $\tilde{\boldsymbol{X}}_i$. To achieve this, we first compute

the hyperedge features of a hypergraph consisting of 4-node hyperedges and then infer the corresponding signatures with a zero-inflated model.

Prediction of hyperedge attributes We consider a hypergraph where each hyperedge groups an individual, a tissue, a metagene, and a cell-type node. For all metagenes $j \in 1..M$, we compute latent hyperedge attributes $\hat{\boldsymbol{e}}_{ij}^{(k,q)}$ for a cell-type q in a tissue of interest k of individual i as follows:

$$\hat{\boldsymbol{e}}_{ij}^{(k,q)} = \mathrm{MLP}(\hat{\boldsymbol{h}}_i^d, \boldsymbol{h}_j^m, \boldsymbol{h}_k^t, \boldsymbol{h}_q^c),$$

where $\boldsymbol{h}_q^c$ are parameters specific to each unique cell-type q and the MLP is shared for all combinations of metagenes, individuals, tissues, and cell-types.

Zero-inflated model We employ the following probabilistic model:

$$\begin{aligned} p(\boldsymbol{x}_i^{(k,q)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k, q) &= \prod_j^G p(x_{ij}^{(k,q)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k, q) \\ p(\boldsymbol{x}_i^{(k,q)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, j, k, q) &= \mathrm{ZINB}(x_{ij}^{(k,q)}; \mu_{ij}^{(k,q)}, \theta_{ij}^{(k,q)}, \pi_{ij}^{(k,q)}), \end{aligned}$$

where ZINB is a zero-inflated negative binomial distribution. The mean $\mu_{ij}^{(k,q)}$, dispersion $\theta_{ij}^{(k,q)}$, and dropout probability $\pi_{ij}^{(k,q)}$ parameters are computed as:

$$\begin{aligned} \boldsymbol{\mu}_i^{(k,q)} &= n_i^{(k,q)} l_i^{(k,q)} \mathrm{softmax}\Big(\boldsymbol{W}_s \hat{\boldsymbol{e}}_i^{(k,q)} + \boldsymbol{b}_s\Big) \\ \boldsymbol{\theta}_i^{(k,q)} &= \exp\Big(\boldsymbol{W}_\theta \hat{\boldsymbol{e}}_i^{(k,q)} + \boldsymbol{b}_\theta\Big) \\ \boldsymbol{\pi}_i^{(k,q)} &= \sigma(\boldsymbol{W}_\pi \hat{\boldsymbol{e}}_i^{(k,q)} + \boldsymbol{b}_\pi), \end{aligned}$$

where $\boldsymbol{W}_s$, $\boldsymbol{W}_\theta$, $\boldsymbol{W}_\pi$, $\boldsymbol{b}_s$, $\boldsymbol{b}_\theta$, and $\boldsymbol{b}_\pi$ are learnable parameters; $n_i^{(k,q)}$ is the number of cells in the signature, and $l_i^{(k,q)}$ is their average library size. At train time, we set $n_i^{(k,q)}$ to match the ground truth number of cells. At test time, the number of cells $n_i^{(k,q)}$ is user-definable. We model the average library size $l_i^{(k,q)}$ with a log-normal distribution

$$\begin{aligned} \log l_i^{(k,q)} &\sim \mathcal{N}(l_i^{(k,q)}; \nu_i^{(k,q)}, \omega_i^{(k,q)}) \\ \nu_i^{(k,q)} &= \boldsymbol{W}_\nu \hat{\boldsymbol{e}}_i^{(k,q)} + \boldsymbol{b}_\nu \\ \omega_i^{(k,q)} &= \exp\Big(\boldsymbol{W}_\omega \hat{\boldsymbol{e}}_i^{(k,q)} + \boldsymbol{b}_\omega\Big), \end{aligned}$$

where $\boldsymbol{W}_\nu$, $\boldsymbol{W}_\omega$, $\boldsymbol{b}_\nu$, and $\boldsymbol{b}_\omega$ are learnable parameters. Optionally, we can use the observed library size.

Optimisation Single-cell transcriptomic studies typically measure single-cell gene expression for a limited number of individuals, tissues, and cell-types, so aggregating single-cell profiles per individual, tissue, and cell-type often results in small sample sizes. To address this challenge, we apply transfer learning by pre-training the hypergraph model f on the multi-tissue imputation task and then fine-tuning the parameters of the signature inference module on the cell-type signature profiles. Concretely, we minimise the loss:

$$\mathcal{L}(\boldsymbol{x}_i^{(k,q)}, \tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i, k, q) = -\log p(\boldsymbol{x}_i^{(k,q)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k, q),$$

which corresponds to the negative log-likelihood of the observed cell-type signatures.

Inference of uncollected gene expression To infer the signature of a cell-type q in a certain tissue v of interest, we first compute the latent individual embeddings $\hat{\boldsymbol{h}}_i^d$ from the multi-tissue profiles $\tilde{\boldsymbol{X}}_i$ and then compute the mean of the distribution $p(\boldsymbol{x}_i^{(k,q)} | \hat{\boldsymbol{h}}_i^d, \boldsymbol{u}_i, k, q)$ as $\boldsymbol{\mu}_i^{(k,q)}(1 - \boldsymbol{\pi}_i^{(k,q)})$. Alternatively, we can draw random samples from that distribution.

5.1.4 Related work

Gene expression imputation methods A standard approach for imputation of uncollected transcriptomics values is to use a proxy tissue (e.g. whole blood) as a surrogate [210]. However, gene expression is known to be tissue and cell-type specific, limiting the effectiveness of this technique. Other related studies infer tissue-specific gene expression from genetic information. [213] propose a mixed-effects model to infer uncollected data in multiple tissues from expression quantitative trait loci (eQTLs). [214] introduce a model termed Meta-Tissue that aggregates information from multiple tissues to increase statistical power of eQTL detection. Nonetheless, these approaches do not model the complex relationships between measured and unmeasured gene expression traits among tissues and cell types, and individual-level genetic information (such as at eQTLs) is often unavailable and subject to privacy concerns. Instead, recent closely related work relies on linear factor analysis and dimensionality reduction techniques. TEEBoT (Tissue Expression Estimation using Blood Transcriptome) [21] projects gene expression from a single reference tissue (i.e. whole blood) into a low-dimensional space via principal component analysis (PCA), followed by linear regression to impute uncollected values. HYFA allows a departure from the linearity assumption of TEEBoT and also handles a variable number of reference tissues.

Knowledge graph embedding techniques Our framework leverages ideas from knowledge graph embedding techniques by using learnable embeddings for biological entities (i.e. tissues, cell-types, and metagenes). Since the advent of word embeddings [224], several approaches have emerged to learn vector representations of entities and relations in knowledge graphs [219, 225–227]. TransE [219] represents entities as low-dimensional vectors and relationships as translations in the embedding space, and optimises parameters through an energy-based objective. TransH [225] extends the TransE framework by projecting the entity embeddings into relation-specific hyperplanes. ComplEx [226] utilises complex vectors that capture antisymmetric entity relations. ConvE [227] models interactions between input entities and relations through convolutional and fully-connected layers. Despite all the recent advances, knowledge graph embeddings have been understudied for modelling higher-order structures (i.e. hyperedges). Moreover, while these methods are capable of link prediction, they are limited to a transductive setting, where the full set of entities (e.g. individuals) must be known at train time [228].

Graph representation learning Graph neural networks remedy this problem by leveraging the structure and properties of graphs to compute node features, allowing to handle unseen entities at inference time (e.g. individuals). Graph neural networks operating on hypergraphs have recently started to flourish, with approaches such as HEAT [229] and *rxn-hypergraph* [230] attaining state-of-the-art results in tasks involving higher-order relationships, such as source code [229] and chemical reactions [230]. In terms of graph-based imputation methods, the closest approach to our framework is GRAPE [218]. GRAPE represents tabular data as a bipartite graph, where observations and features are two types of nodes, and the observed feature values are attributed edges between the nodes [218]. Imputation of missing features then corresponds to an edge-level prediction task. HYFA subsumes GRAPE's bigraph in that our hypergraph becomes a bipartite graph when a single metagene is employed. This allows for a trade-off between feature granularity and over-squashing, which happens when information from a large receptive field is compressed into fixed-length node vectors [220]. In terms of message passing, HYFA distinguishes between dynamic nodes (udpated during message passing) and static nodes (with learnable node features that are not updated during message passing), eliminating the dependency of tissue and metagene representations on donor features and, by transitivity, undesired dependencies between inviduals. HYFA is thus a hybrid and flexible approach that combines features from knowledge graph embedding and graph representation learning techniques.

Single-cell variational inference Our framework is related to single-cell variational inference (scVI) [94] in that it can also be optimised via variational inference (Supplementary Information D.5), e.g. via a (zero-inflated) negative binomial likelihood, treating the individual representations as latent variables. In contrast to scVI, however, HYFA offers features to handle a variable number of reference tissues. It also incorporates inductive biases to reuse knowledge across tissues, allowing the model to scale to larger multi-tissue samples.

5.1.5 eQTL mapping

The breadth of tissues in the GTEx (v8) collection enables us to comprehensively evaluate the extent to which eQTL discovery could be improved through the HYFA-imputed transcriptome data. We map eQTLs that act in cis to the target gene (cis-eQTLs), using all SNPs within $\pm$ 1 Mb of the transcription start site of each gene. For the imputed and the original (incomplete) datasets, we consider SNPs significantly associated with gene expression, at a false discovery rate ≤ 0.10. We apply the same GTEx eQTL mapping pipeline, as previously described [48], to the imputed and original datasets to quantify the gain in eQTL discovery from the HYFA-imputed dataset.

eQTL mapping In Chapter 2 (Section 2.2.4), we review a standard eQTL mapping approach.

5.1.6 GTEx bulk and single-nucleus RNA-seq data processing

The GTEx dataset is a public resource that has generated a broad collection of gene expression data collected from a diverse set of human tissues [12]. We downloaded the data from the GTEx portal. After the processing step, the GTEx-v8 dataset consisted of 15197 samples (49 tissues, 834 donors) and 12557 genes. The dataset was randomly split into 500 train, 167 validation, and 167 test donors. Each donor had an average of 18.22 collected tissues. The processing steps are described below.

Normalised bulk transcriptomics (GTEx-v8) Following the GTEx eQTL discovery pipeline (`https://github.com/broadinstitute/gtex-pipeline/tree/master/qtl`), we processed the data as follows:

1. Discard underrepresented tissues (n=5), namely bladder, cervix (ectocervix, endocervix), fallopian tube, and kidney (medulla). Discard donors with only one collected tissue (n=4).

2. Select set of overlapping genes across all tissues. Select genes based on expression thresholds of ≥ 0.1 transcripts per kilobase million (TPM) in $\geq 20\%$ of samples and ≥ 6 reads (unnormalised) in $\geq 20\%$ of samples.

3. Normalise read counts across samples using the trimmed mean of M values (TMM) method [59].

4. Apply inverse normal transformation to the expression values for each gene.

Cell-type signatures from a paired snRNA-seq dataset (GTEx-v9) We downloaded paired snRNA-seq data for 16 GTEx individuals [215] collected in 8 GTEx tissues, namely skeletal muscle, breast, esophagus (mucosa, muscularis), heart, lung, prostate, and skin. We split these individuals into train, validation, and test donors according to the GTEx-v8 split. We processed the data as follows:

1. Select set of overlapping genes between bulk RNA-seq (GTEx-v9) and paired snRNA-seq dataset [215].

2. Select top 3000 variable genes using the function `pp.highly_variable_genes` from the Scanpy library [231] with flavour setting `seurat_v3` [232].

3. Discard underrepresented cell-types occurring in less than 10 tissue-individual combinations.

4. Aggregate (i.e. sum) read counts by individual, tissue, and (broad) cell-type. This resulted in a dataset of 226 unique signatures, of which 135 belong to matching GTEx-v8 individuals.

5.2 Results

Hypergraph factorisation (HYFA) We developed HYFA, a framework for inferring the transcriptomes of unmeasured tissues and cell-types from bulk expression collected in a variable number of reference tissues (Figure 5.2).

HYFA receives as input gene expression measurements collected from a set of reference tissues, as well as demographic information, and outputs gene expression values in a tissue of interest (e.g. uncollected). The first step of the workflow is to project the input gene expression into low-dimensional *metagene* representations [216, 217] for every collected tissue. Each metagene summarises abstract properties of groups of genes, e.g. sets of genes that tend to be expressed together [233], that are relevant for

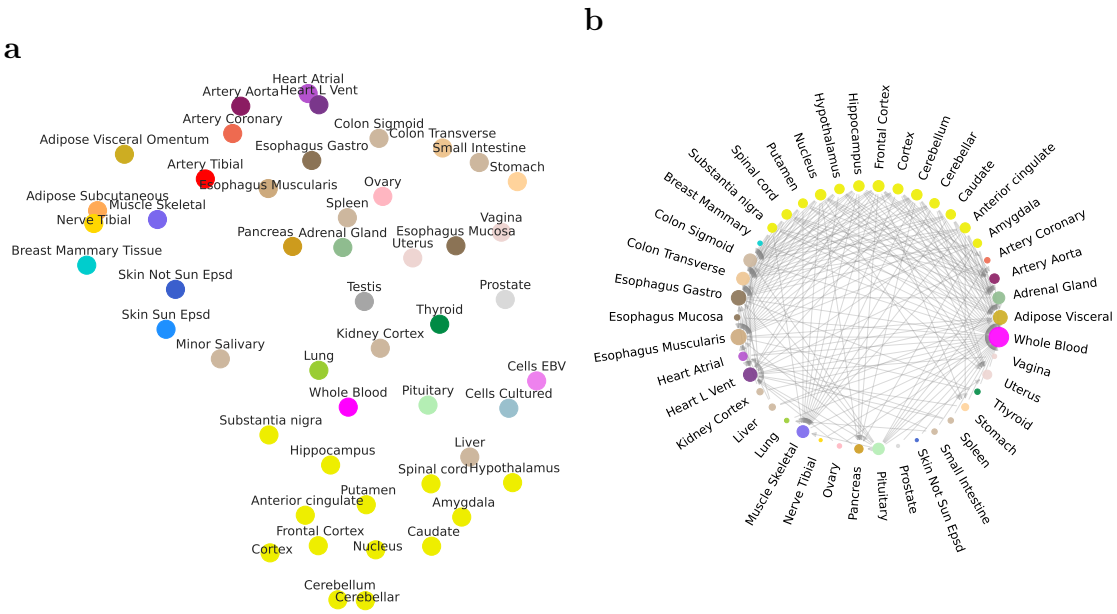


Fig. 5.3 Analysis of cross-tissue relationships. Colors are assigned to conform to the GTEx Consortium conventions. (a) UMAP representation of the tissue embeddings learnt by HYFA. Note that human body systems cluster in the embedding space (e.g. digestive system: stomach, small intestine, colon, esophagus; and central nervous system). (b) Network of tissues depicting the predictability of target tissues with HYFA using the average per-sample Pearson ρ correlation coefficients. The dimension of each node is proportional to its degree. Edges from reference to target tissues indicate an average Pearson correlation coefficient $\rho > 0.5$. Interestingly, central nervous system tissues strongly correlate with several non-brain tissues such as gastrointestinal tissues.

the imputation task. In a second step, HYFA employs a custom message passing neural network [234] that operates on a 3-uniform hypergraph, yielding factorised individual, tissue, and metagene representations. Lastly, HYFA infers latent metagene values for the target tissue — a hyperedge-level prediction task — and maps these representations back to the original gene expression space. Through higher-order hyperedges (e.g. 4-uniform hypergraph), HYFA can also incorporate cell-type information and infer finer-grained cell-type-specific gene expression.

Altogether, HYFA offers features to reuse knowledge across tissues and genes, capture non-linear cross-tissue patterns of gene expression, learn rich representations of biological entities, and account for variable numbers of reference tissues.

Characterisation of cross-tissue relationships Characterising cross-tissue relationships at the transcriptome level can help elucidate coordinated gene regulation and

expression, a fundamental phenomenon with direct implications on health homeostasis, disease mechanisms, and comorbidities [235–237].

We trained HYFA on bulk gene expression from the GTEx project (GTEx-v8) [12] and assessed the cross-tissue gene expression predictability —measured by the Pearson correlation between the observed and the predicted gene expression across individuals— and quality of tissue embeddings (Figure 5.3). Application of Uniform Manifold Approximation and Projection (UMAP) [238] on the learnt tissue representations revealed strong clustering of biologically-related tissues (Figure 5.3a), including the gastrointestinal system (e.g. esophageal, stomach, colonic, and intestinal tissues), the female reproductive tissues (i.e. uterus, vagina, and ovary), and the central nervous system (i.e. the 13 brain tissues). The clustering properties were robust across UMAP runs and could be similarly appreciated using other dimensionality reduction algorithms such as t-distributed Stochastic Neighbor Embedding (t-SNE) [239].

For every pair of reference and target tissues in GTEx, we then computed the Pearson correlation coefficient ρ between the predicted and actual gene expression, averaged the scores across individuals, and used a cutoff of $\rho > 0.5$ to depict the top pairwise associations (Figure 5.3b and Supplementary Information D.8). We observed connections between most GTEx tissues and whole blood, which suggests that blood-derived gene expression is highly informative of (patho)physiological processes in other tissues [240]. Notably, brain tissues and the pituitary gland were strongly associated with several tissues ($\rho > 0.5$), including gastrointestinal tissues (e.g. esophagus, stomach, and colon), the adrenal gland, and skeletal muscle, which may account for known disease comorbidities.

Imputation of gene expression from whole blood transcriptome Knowledge about tissue-specific patterns of gene expression can increase our understanding of disease biology, facilitate the development of diagnostic tools, and improve patient subtyping [241, 21], but most tissues are inaccessible or difficult to acquire.

To address this challenge, we studied to what extent HYFA can recover tissue-specific gene expression from whole-blood transcriptomic measurements (Figures 5.4 and 5.5).

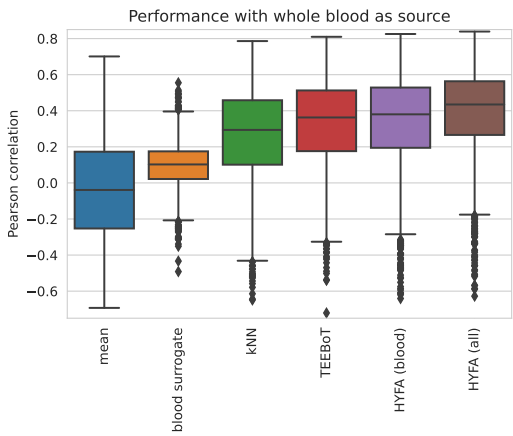


Fig. 5.5 Prediction performance from whole-blood gene expression (n=2424 samples from 167 test GTEx donors).

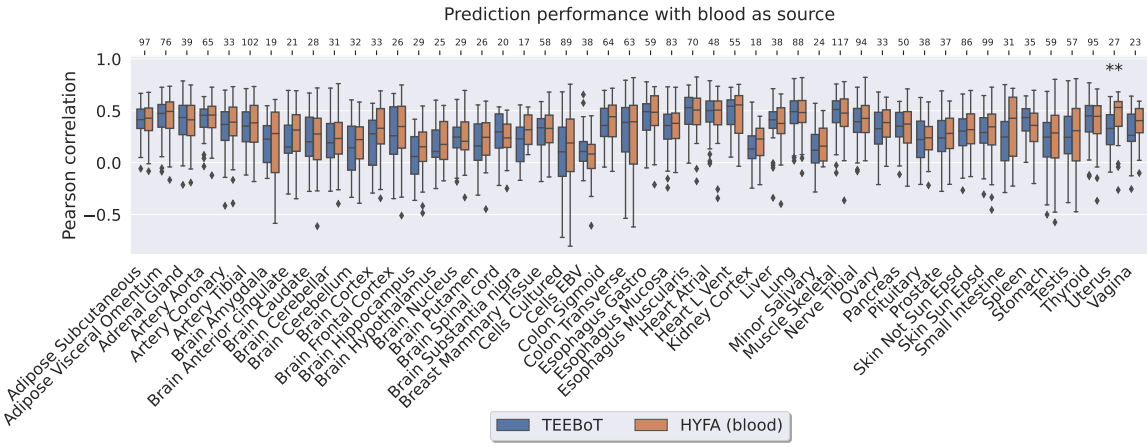


Fig. 5.4 Performance comparison across gene expression imputation methods. Per-tissue comparison between HYFA and TEEBoT when using whole-blood and as reference. HYFA achieved superior Pearson correlation in 25 out of 48 target tissues when a single tissue was used as reference. We employed a two-sided Mann-Whitney-Wilcoxon tests to compute p-values (*: 1e-2 $<$ p $\leq$ 5e-2, **: 1e-3 $<$ p $\leq$ 1e-2, ***: 1e-4 $<$ p $\leq$ 1e-3, ****: p $\leq$ 1e-4). Boxes show quartiles, centerlines correspond to the median, and whiskers depict the distribution range (1.5 times the interquartile range). Outliers outside of the whiskers are shown as distinct points. The top axis indicates the total number n of independent individuals for every target tissue.

For each test individual with measured whole-blood gene expression, we predicted tissue-specific gene expression in the remaining collected tissues of the individual. We evaluated performance using the Pearson correlation ρ between the inferred gene expression and the ground-truth samples. We observed strong prediction performance for esophageal tissues (muscularis: $\rho = 0.49$, gastro: $\rho = 0.46$, mucosa: $\rho = 0.36$), heart tissues (left ventricle: $\rho = 0.48$, atrial: $\rho = 0.46$), and lung ($\rho = 0.47$), while Epstein Barr virus-transformed lymphocytes ($\rho = 0.06$), an accessible and renewable resource for functional genomics, was a notable outlier. We compared our method with the following baselines:

- **Mean imputation**: Replaces missing values with the feature averages.
- **Blood surrogate**: Utilises expression in blood as a proxy for the target tissue.
- **k-Nearest Neighbours (kNN)**: Imputes missing features with the average of measured values across the k nearest observations (k=20).
- **TEEBoT without single-nucleotide polymorphism information** [21]: Projects the high-dimensional blood expression data into a low-dimensional

space through principal component analysis (30 components; 75-80% explained variance) and then performs linear regression to predict the gene expression of the target tissue.

- **HYFA (all)**: Employs information from all collected tissues of the individual.

Overall, TEEBoT and HYFA attained comparable scores when a single tissue (i.e. whole blood) was used as reference and both methods outperformed standard imputation approaches (mean imputation, blood surrogate, and k nearest neighbours; Figure 5.5).

The blood-imputed gene expression also predicted disease-relevant genes in hard-to-access central nervous system (Supplementary Information D.10). These include *APP*, *PSEN1*, and *PSEN2*, i.e. the causal genes for autosomal dominant forms of early-onset Alzheimer's disease [242], and Alzheimer's disease genetic risk factors such as *APOE* [243]. We noted that the per-gene prediction scores followed smooth distributions (Supplementary Information D.9).

Multiple reference tissues improve performance We hypothesised that using multiple tissues as reference would improve downstream imputation performance. To evaluate this, we selected individuals with measured gene expression both at the target tissue and 4 reference accessible tissues (whole blood, skin sun-exposed, skin not sun-exposed, and adipose subcutaneous) and employed HYFA to impute target expression values (Figures 5.6 and 5.7, and Supplementary Information D.12). We discarded target tissues with less than 25 test individuals.

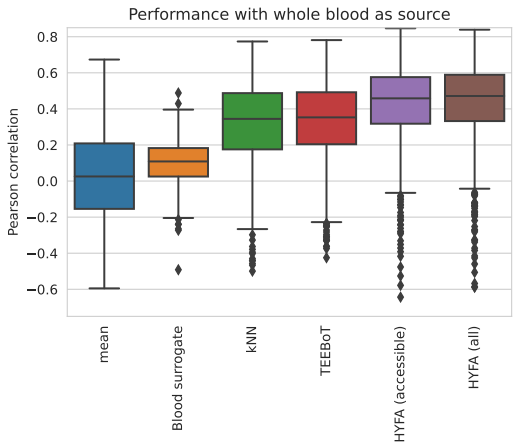


Fig. 5.6 Prediction performance from accessible tissues as reference (n=675 samples from 167 test GTEx donors).

Relative to using whole blood in isolation, using all accessible tissues as reference resulted in improved performance for 32 out of 38 target tissues (Supplementary Information D.11). This particularly boosted imputation performance for esophageal tissues (muscularis: $\Delta\rho = 0.068$, gastro: $\Delta\rho = 0.061$, mucosa: $\Delta\rho = 0.048$), colonic tissues (transverse: $\Delta\rho = 0.065$, sigmoid: $\Delta\rho = 0.056$), and artery tibial ($\Delta\rho = 0.079$). In contrast, performance for the pituitary gland ($\Delta\rho = -0.011$), lung ($\Delta\rho = -0.003$), and stomach ($\Delta\rho = -0.002$) remained stable or dropped slightly. Moreover, the performance gap between HYFA and TEEBoT (trained on the set of complete multi-tissue samples) widened relative to the single-tissue scenario (Figures 5.6 and 5.7) —

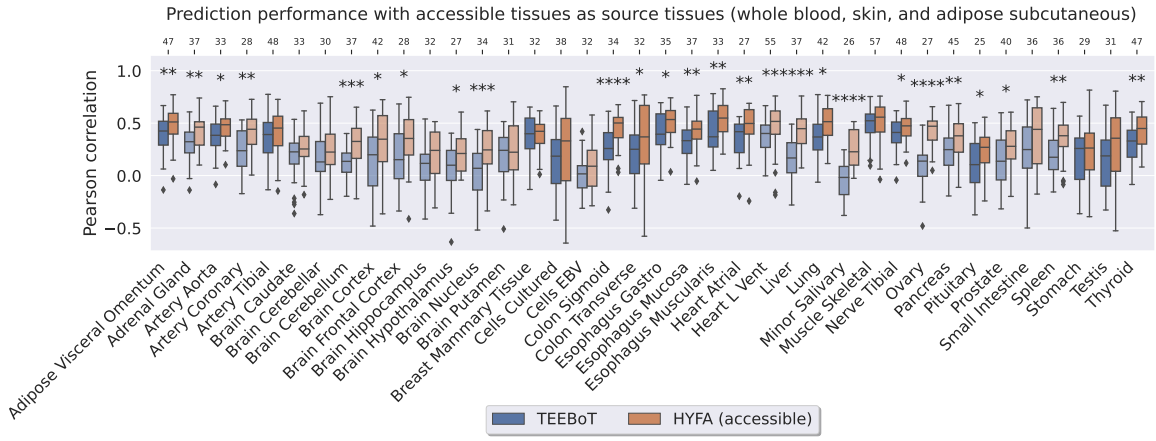


Fig. 5.7 Per-tissue comparison between HYFA and TEEBoT when using all accessible tissues (whole blood, skin sun exposed, skin not sun exposed, and adipose subcutaneous) as reference. HYFA achieved superior Pearson correlation in all target tissues when multiple reference tissues were considered. For underrepresented target tissues (less than 25 individuals with source and target tissues in the test set), we considered all the validation and test individuals (translucent bars). We employed a two-sided Mann-Whitney-Wilcoxon tests to compute p-values (*: 1e-2 $< p \leq$ 5e-2, **: 1e-3 $< p \leq$ 1e-2, ***: 1e-4 $< p \leq$ 1e-3, ****: $p \leq$ 1e-4). Outliers outside of the whiskers are shown as distinct points. The top axis indicates the total number n of independent individuals for every target tissue.

HYFA obtained better performance in all target tissues, with statistically significant improvements in 26 out of 38 tissues (two-sided Mann-Whitney-Wilcoxon p-value $<$ 0.05). We attribute the improved scores to HYFA's ability to process a variable number of reference tissues, reuse knowledge across tissues, and capture non-linear patterns.

Inference of cell-type signatures We next investigated the potential of HYFA to predict cell-type-specific signatures — average gene expression across cells from a given cell-type — in a given tissue of interest. We first selected GTEx donors with collected bulk (v8) and single-nucleus RNA-seq profiles (v9). Next, we trained HYFA to infer cell-type signatures from the multi-tissue bulk expression profiles. We evaluated performance using the observed (Figure 5.8) and inferred library sizes (Supplementary Information D.17). To attenuate the small data size problem, we applied transfer learning on the model trained for the multi-tissue imputation task.

We observed strong prediction performance (Pearson correlation ρ between log ground truth and log predicted signatures) for vascular endothelial cells (heart: $\rho = 0.84$; breast: $\rho = 0.88$, esophagus muscularis: $\rho = 0.68$) and fibroblasts (heart: $\rho = 0.84$; breast: $\rho = 0.89$, esophagus muscularis: $\rho = 0.70$). Strikingly, HYFA recovered the cell-

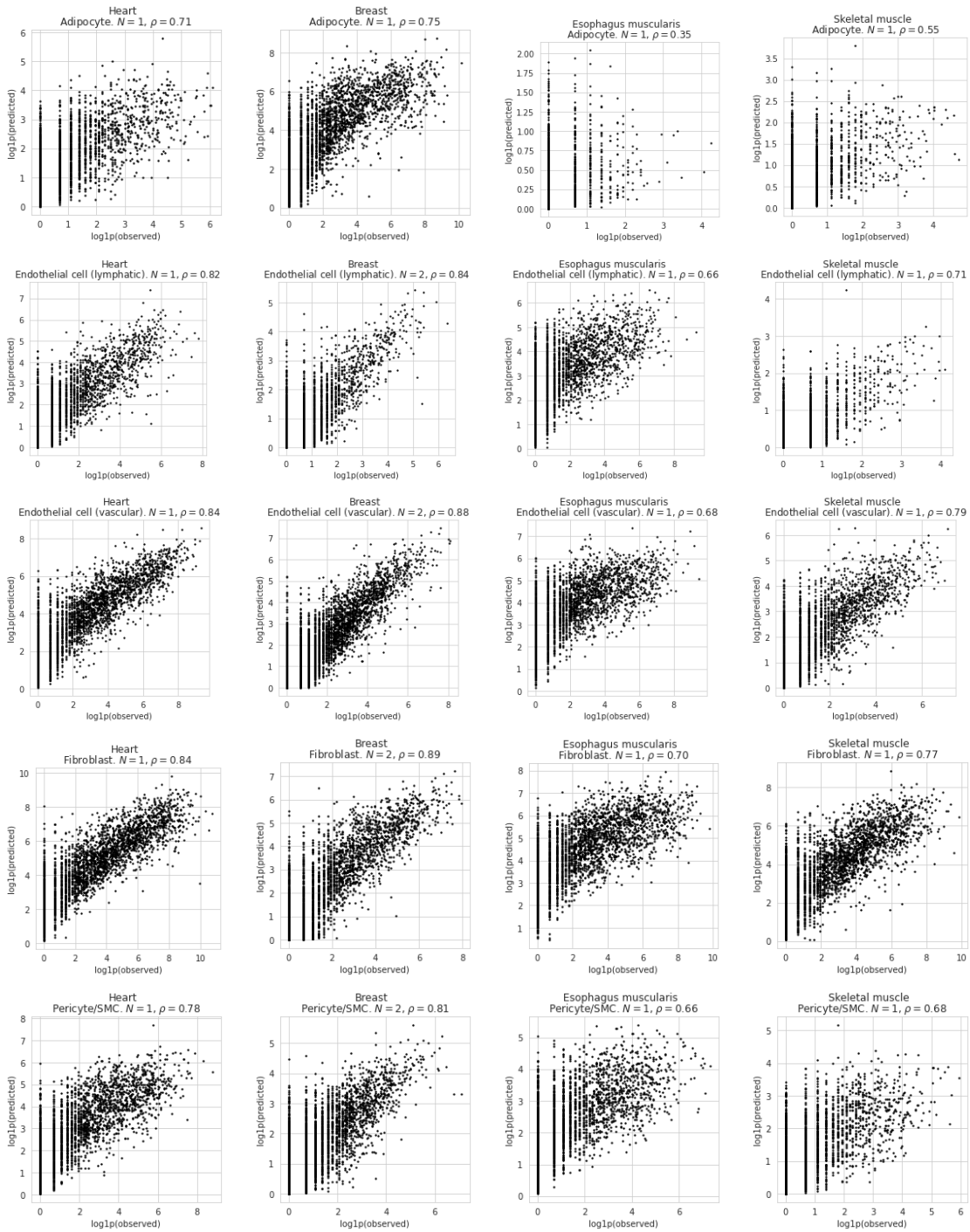


Fig. 5.8 Prediction of cell-type signatures. HYFA imputes individual- and tissue-specific cell-type signatures from bulk multi-tissue gene expression. The scatter plots depict the Pearson correlation ρ between the logarithmised ground truth and predicted signatures for N unseen individuals. The model never observes skeletal muscle signatures at transfer time. To infer the signatures, we used the observed library size $l_i^{(k,q)}$ and number of cells $n_i^{(k,q)}$.

type profiles of tissues that were never observed in the train set with high correlation (Figure 4 and Supplementary Information K), e.g. skeletal muscle (vascular endothelial cells: $\rho = 0.79$, fibroblasts: $\rho = 0.77$, pericytes/SMC: $\rho = 0.68$), demonstrating the benefits of the factorised tissue representations.

Overall, our results highlight the potential of HYFA to impute unknown cell-type signatures even for tissues that were not considered in the original single-cell study. In the future, as single-cell RNA-seq datasets become larger in number of individuals, we hypothesise that the resolution of HYFA's inferred signatures will increase, with possible benefits in terms of downstream analyses. Our analyses point to promising downstream applications as single-cell RNA-seq datasets become larger in number of individuals (Supplementary Information D.19), including deconvolution and cell-type specific eQTL mapping.

eQTL mapping The breadth of tissues in the GTEx (v8) collection enables us to comprehensively evaluate the extent to which eQTL discovery could be improved through the HYFA-imputed transcriptome data. We map eQTLs that act in cis to the target gene (cis-eQTLs), using all SNPs within $\pm$ 1 Mb of the transcription start site of each gene. For the imputed and the original (incomplete) datasets, we consider SNPs significantly associated with gene expression, at a false discovery rate ≤ 0.10. We apply the same GTEx eQTL mapping pipeline, as previously described [48], to the imputed and original datasets to quantify the gain in eQTL discovery from the HYFA-imputed dataset. In Chapter 2 (Section 2.2.4), we review the intuition behind eQTL mapping.

Multi-tissue imputation improves eQTL detection Gene expression acts as an intermediate molecular trait between DNA and phenotype and, therefore, genetic mapping of genome-wide gene expression can shed light on the genetic architecture and molecular basis of complex traits. The GTEx project has enabled the identification of numerous genetic associations with gene expression across a broad collection of tissues [12], also known as expression Quantitative Trait Loci (eQTLs) [248]. However, eQTL datasets are characterised by small sample sizes, especially for difficult-to-acquire tissues and cell types, reducing the statistical power to detect eQTLs [249].

To address this problem, we employed HYFA to impute the transcript levels of every uncollected tissue for each individual in GTEx, yielding a complete gene expression dataset of 834 individuals and 49 tissues. We then performed eQTL mapping on the original and imputed datasets and observed a substantial gain in the number of unique

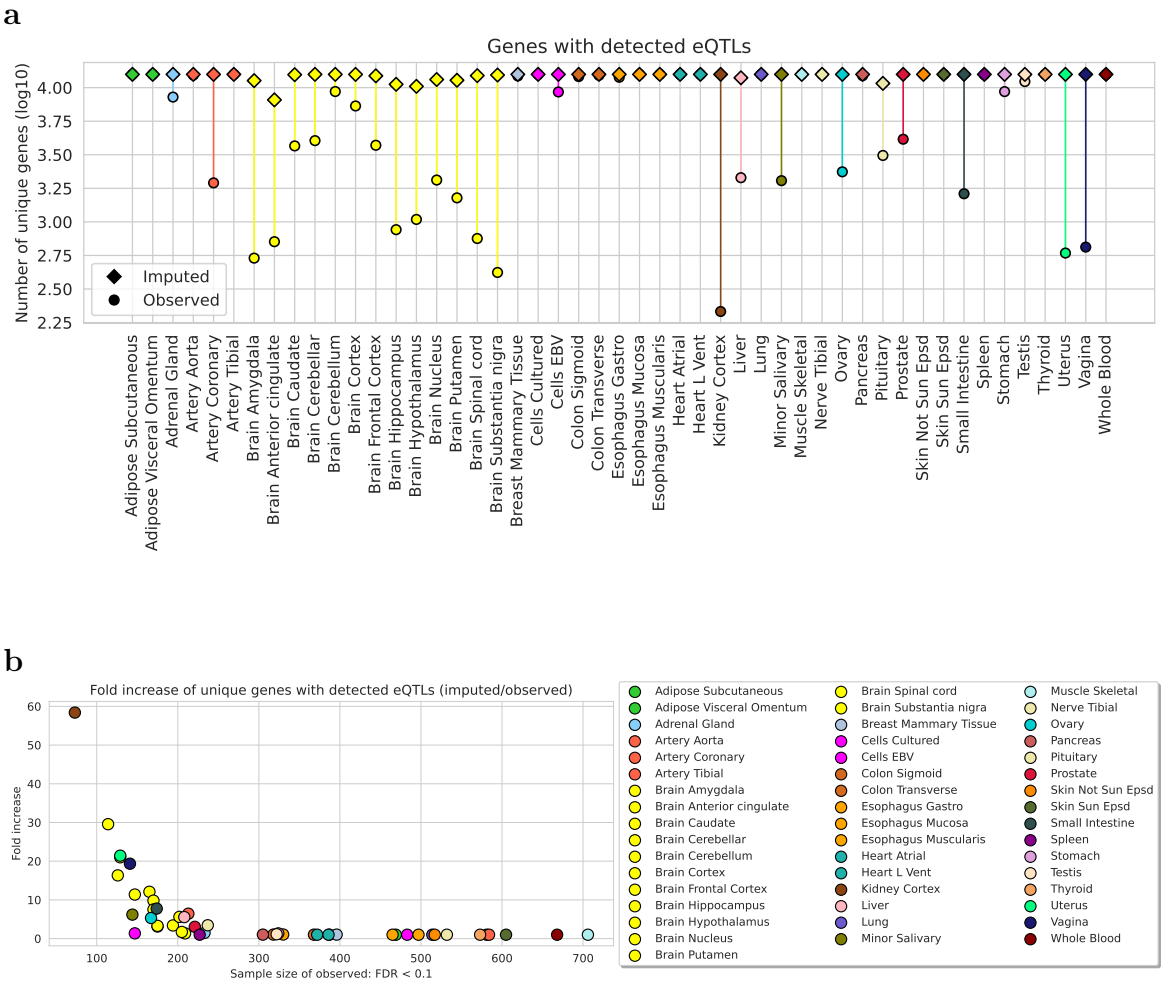


Fig. 5.9 HYFA's imputed data improves expression Quantitative Trait Loci (eQTL) discovery. (a) Number of unique genes with detected eQTLs (FDR < 0.1) on observed (circle) and full (observed plus imputed; rhombus) GTEx data. Note logarithmic scale of y-axis. The eQTLs were mapped using MatrixEQTL [86, 48] assuming additive genotype effect on gene expression. MatrixEQTL conducts a test for each SNP-gene pair and makes adjustments for multiple comparisons by computing the Benjamini-Hochberg FDR [244]. (b) Fold increase in number of unique genes with mapped eQTLs (y-axis) versus observed sample size (x-axis).

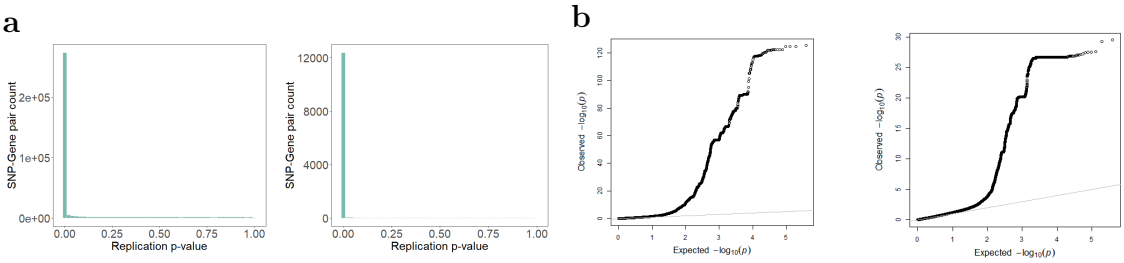


Fig. 5.10 HYFA recovers replicable and experimentally validated expression Quantitative Trait Loci (eQTLs). (a) Histogram of replication p-values among the HYFA-identified cis-eQTLs for whole blood (left) and brain prefrontal cortex (right). For replication, we used the independent eQTLGen Consortium ($n > 30,000$; [245]) and PsychENCODE ($n = 1,866$; [246]) eQTL datasets, respectively. (b) Quantile-quantile plot showing the causal variants' association with gene expression in blood (left) and brain frontal cortex (right) in the HYFA-derived dataset using experimentally validated causal variant data from the application of Massively Parallel Reporter Assay [247]. All statistical tests were two-sided. HYFA's imputed data substantially increases the number of identified associations with high replicability and significant enrichment of causal regulatory variants.

genes with detected eQTLs, the so-called *eGenes* (Figure 5.9). Notably, this metric increased for tissues with low sample size (Spearman correlation coefficient $\rho = -0.83$) — which are most likely to benefit from borrowing information across tissues with shared regulatory architecture. Kidney cortex displayed the largest gain in number of eGenes (from 215 to 12,557), while there was no increase observed for whole blood.

To assess the quality of the identified eQTLs from HYFA imputation, we conducted systematic replication analyses of 1) the whole blood eQTL-eGene pairs, using the eQTLGen blood transcriptome dataset in more than 30,000 individuals [245] and 2) the frontal cortex eQTL-eGene pairs, using the PsychENCODE pre-frontal cortex transcriptome dataset in 1,866 individuals [246]. For each tissue, we quantified the replication rate for eQTL-eGene pairs using the π_1 statistic [250]. Notably, we found a highly significant enrichment for low replication p-values among the HYFA-derived eQTL-eGene pairs (Figure 5.10), demonstrating strong reproducibility of the results. The replication rate π_1 was 0.80 for whole blood and 0.96 for frontal cortex. We also evaluated the extent to which the HYFA imputation captured regulatory variants that directly modulate gene expression using experimentally validated causal variants from Massively Parallel Reporter Assay [247]. Notably, among the causal regulatory variants

from this experimental assay, we found a highly significant enrichment for low p-values among the HYFA-identified eQTLs in blood and in frontal cortex (Figure 5.10).

Thus, HYFA imputation enabled identification of biologically meaningful, replicable eQTL hits in the respective tissues. Our results generate a large catalog of new tissue-specific eQTLs, with potential to enhance our understanding of how regulatory variation mediates variation in complex traits, including disease susceptibility.

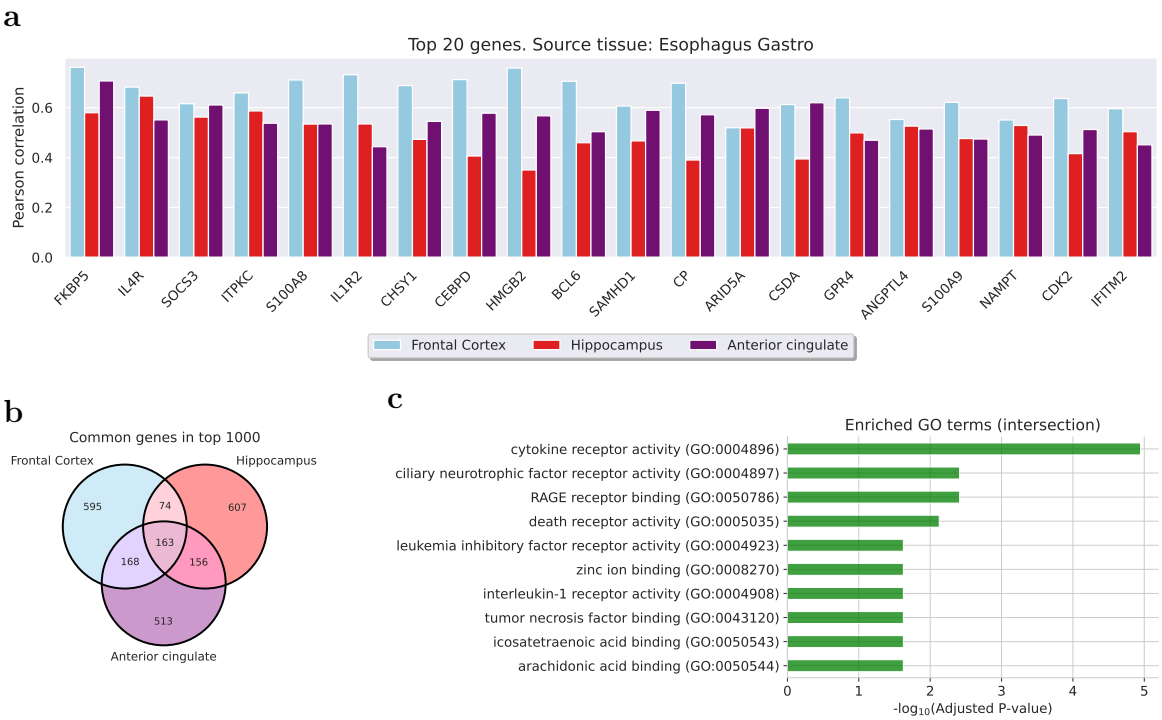


Fig. 5.11 Top predicted genes in multiple brain regions with the oesophagogastric junction as the reference tissue. (a) Top predicted genes in multiple brain regions with the oesophagogastric junction as the reference tissue, ranked by average Pearson correlation. (b) Common genes in the top 1000 predicted genes for each brain tissue. (c) Enriched GO terms of the top shared genes at the interesection. The top predicted genes were enriched in signalling pathways (FDR < 0.05), consistent with studies reporting that gut microbes communicate to the central nervous system through endocrine and immune mechanisms. These results depict the cross-tissue associations and highlight the potential connection between the elements of the oesophagogastric junction and the ciliary neurotrophic factor, which has been linked to the survival of neurons [251] and the control of body weight [252].

Brain-gut axis The brain-gut axis is a bidirectional communication system of signalling pathways linking the central and enteric nervous systems. We investigated

the extent to which the transcriptomes of tissues from the gastrointestinal system are predictive of gene expression in brain tissues.

We selected all the unseen individuals with simultaneous measurements in gastrointestinal tissues (i.e. oesophagogastric junction) and brain tissues (i.e. frontal cortex, hippocampus, and anterior cingulate) and employed HYFA to predict the expression values of brain tissues (Figure 5.11). We observed a small number of individuals with measurements in both brain and non-brain tissues (Supplementary Information D.7). After ranking the genes according to their prediction scores and selecting the top 1000 genes for each brain tissue (Venn diagram; Figure 5.11b), we found considerable overlap between the 3 brain tissues (153 common genes in the intersection).

We then used Enrichr [253] with the gene sets *GO_Biological_Process_2021* and *GO_Molecular_Function_2021* to identify the enriched Gene Ontology (GO; [79]) terms for the shared genes at the intersection (Figure 5.11c). Overall, the top predicted genes were enriched in multiple signalling-related terms (e.g. cytokine receptor activity and interleukin-1 receptor activity). This aligns with studies that highlight that gut microbes communicate with the central nervous system through endocrine and immune signalling mechanisms [254]. Genes in the intersection were also notably enriched in the ciliary neurotrophic factor receptor activity (molecular function), which plays an important role in the survival of neurons [251], the development of the enteric nervous system [255], and the control of body weight [252]. Moreover, our results suggest an association with the Receptor for Advanced Glycation Endproducts (RAGE), which has been linked to inflammation-related pathological states such as vascular disease, diabetes, and neurodegeneration [256].

HYFA-learned metagenes capture known biological pathways A key feature of HYFA is that it reuses knowledge across tissues and metagenes, allowing to exploit shared regulatory patterns. We explored whether HYFA's inductive biases encourage learning biologically relevant metagenes. To determine the extent to which metagene-factors relate to known biological pathways, we applied Gene Set Enrichment Analysis (GSEA) [85] to the gene loadings of HYFA's encoder. Similar to [257], for a given query gene set, we calculated the maximum running sum of enrichment scores by descending the sorted list of gene loadings for every metagene and factor. We then computed pathway enrichment p-values through a permutation test and employed the Benjamini-Hochberg method to correct for multiple testing independently for every metagene-factor.

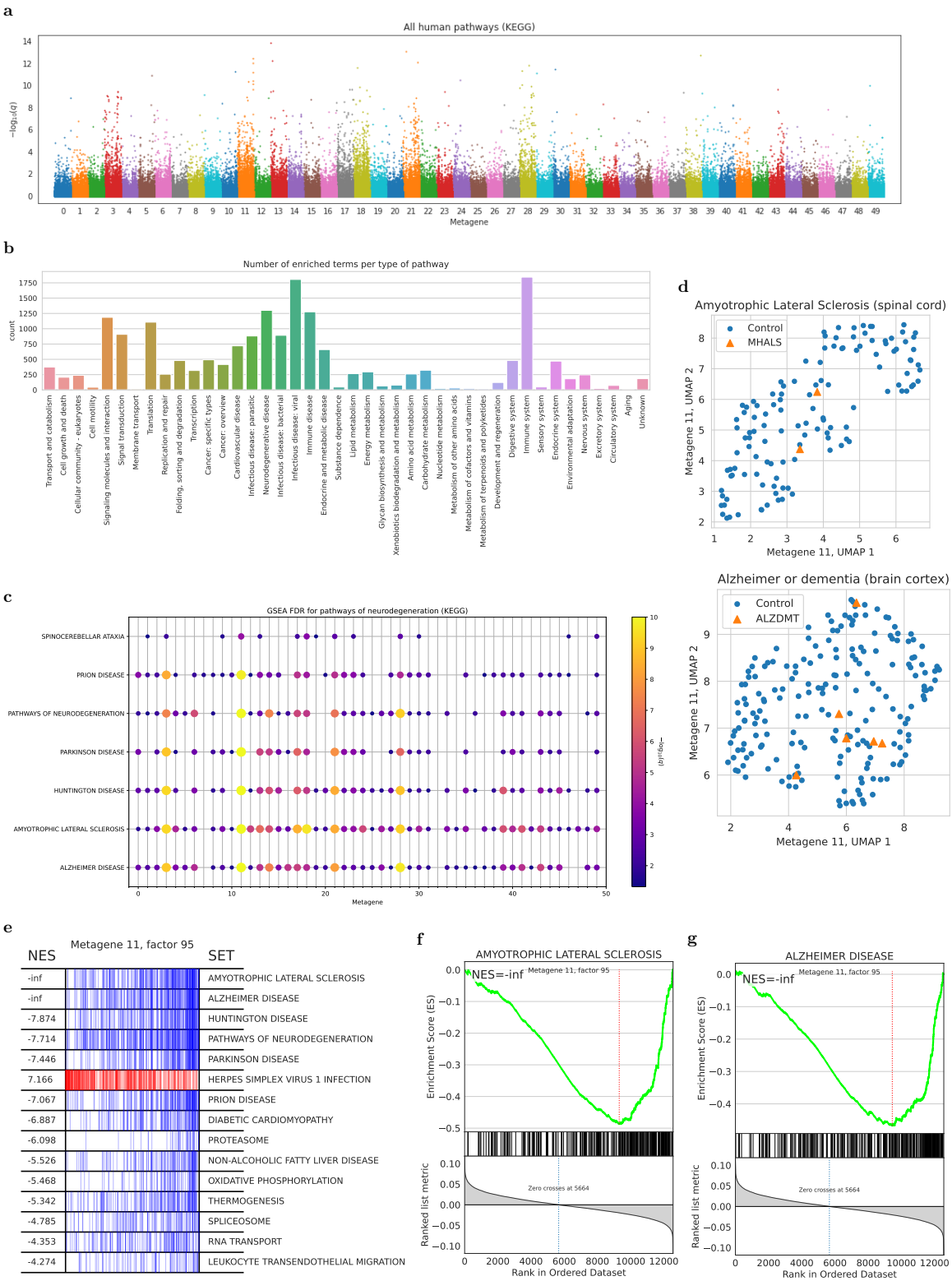


Fig. 5.12 Pathway enrichment analysis of metagene factors (*next page*).

Fig. 5.12 *(previous page)* Pathway enrichment analysis of metagene factors. (a) Manhattan plot of the GSEA results on the metagenes (n=50) and factors (n=98) learned by HYFA. The x-axis represents metagenes (colored bins) and every offset within the bin corresponds to a different factor. The y-axis is the –log q-value (FDR) from the GSEA permutation test, corrected for multiple testing via the Benjamini-Hochberg procedure. We identified 18683 statistically significant enrichments (FDR $<$ 0.05) of KEGG biological processes across all metagenes and factors. (b) Total number of enriched terms for each type of pathway. (c) FDR for pathways of neurodegeneration. For every pathway and metagene, we selected the factor with lowest FDR and depicted statistically significant values (FDR $<$ 0.05). Point sizes are proportional to $-\log$ FDR values. Metagene 11 (factor 95) had the lowest FDR for both Amyotropic Lateral Sclerosis (ALS) and Alzheimer's Disease. (d) UMAP of latent values of metagene 11 for all spinal cord (ALS: orange) and brain cortex (Alzheimer's disease or Dementia: orange) GTEx samples. (e) Leading edge subsets of top-15 enriched gene sets for factor 95 of metagene 11. (f, g) Enrichment plots for Amyotropic Lateral Sclerosis (f) and Alzheimer's disease gene sets (g).

In total, we identified 18683 statistically significant enrichments (FDR $<$ 0.05) of KEGG biological processes ([82]; 320 gene sets; Figure 6) across all HYFA metagenes (n=50) and factors (n=98). Among the enriched terms, 2109 corresponded to signalling pathways and 1300 to pathways of neurodegeneration. We observed considerable overlap between several metagenes in terms of biologically related pathways, e.g. factor 95 of metagene 11 had the lowest FDR for both Alzheimer's disease (FDR $<$ 0.001) and Amyotrophic Lateral Sclerosis (FDR $<$ 0.001) pathways. Enrichment analysis of TRRUST [258] transcription factors (TFs; Supplementary Information D.13) further identified important regulators (Figure 5.13) including GATA1 (known to regulate the development of red blood cells [259]), SPI1 (which controls hematopoietic cell fate [260]), CEBPs (which play an important role in the differentiation of a range of cell types and the control of tissue-specific gene expression; [261, 262]), and STAT1 (a member of the STAT family that drives the expression of many target genes [263]).

We also observed that the learnt HYFA factors recapitulate synergistic effects among the enriched TFs (Figure 5.14 and Supplementary Information D.13). For example, GATA1 and SPI1, which were simultaneously enriched in 7 factors (FDR $<$ 0.05), functionally antagonise each other through physical interaction [264]. Similarly, IRF1 induces STAT1 activation via phosphorylation [263, 265] and both TFs were enriched in 10 factors (FDR $<$ 0.05), aligning with our enrichment analyses of GO Biological Process terms (Supplementary Information D.14). We observed highly specific HYFA factor - TF associations, e.g. GATA1 was enriched (FDR $<$ 0.05) in

factor 69 of 28 out of 50 metagenes (Figure 5.15). Altogether, our analyses suggest that HYFA-learned metagenes and factors are amenable to biological interpretation and capture information about known regulators of tissue-specific gene expression.

Pathway enrichment analysis Similar to [257], we employ Gene Set Enrichment Analysis (GSEA) [85] to relate HYFA's metagene factors to known biological pathways. This is advantageous to over-representation analysis (Chapter 2, section 2.2.3), which requires selecting an arbitrary cutoff to select enriched genes. GSEA, instead, computes a running sum of enrichment scores by descending a sorted gene list [85, 257].
We apply GSEA to the gene loadings in HYFA's encoder. Specifically, let $\boldsymbol{W}_j \in \mathbb{R}^{F \times G}$ be the gene loadings for metagene j, where F is the number of factors (i.e. number of hyperedge attributes) and G is the number of genes (Equation 5.1). For every factor in $\boldsymbol{W}_j$, we employ `blitzGSEA` [266] to calculate the running sum of enrichment scores by descending the gene list sorted by the factor's gene loadings. The enrichment score for a query gene set is the maximum difference between $p_{hit}(\mathcal{S}, i)$ and $p_{miss}(\mathcal{S}, i)$ [257], where $p_{hit}(\mathcal{S}, i)$ is the proportion of genes in $\mathcal{S}$ weighted by their gene loadings up to gene index i in the sorted list [257]. We then calculate pathway enrichment p-values through a permutation test (with n=100 trials) by randomly shuffling the gene list. We use the Benjamini-Hochberg method to correct for multiple testing.

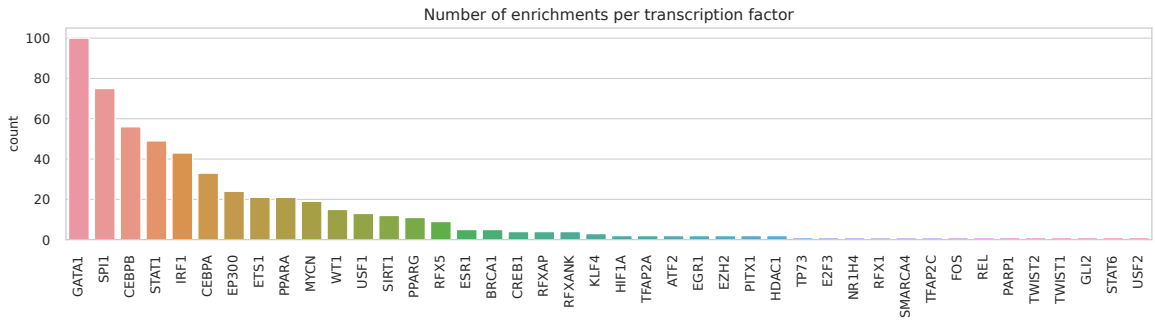


Fig. 5.13 Top enriched transcription factors (TFs), ranked by the total number of metagene-factors in which the TFs were enriched (FDR < 0.05). For every metagene (n=50) and factor (n=98), we performed Gene Set Enrichment Analysis using the corresponding gene loadings of HYFA's encoder and TF gene sets from the TRRUST database of transcription factors (Enrichr library: *TRRUST_Transcription_Factors_2019*.

Fig. 5.14 Circos plot of the top 9 enriched TFs (outer layer). The angular size is proportional to the number of enrichments. The second layer (bar plot) depicts the factor IDs where the TF was enriched, ranging from 0 (lowest bar) to 98 (higher bar). The third layer shows the corresponding metagene IDs (blue dots) of the enriched metagene-factors, increasing monotonically within the same factor. The edges in the middle connect TFs whenever they are both enriched in the same factor (FDR < 0.05).

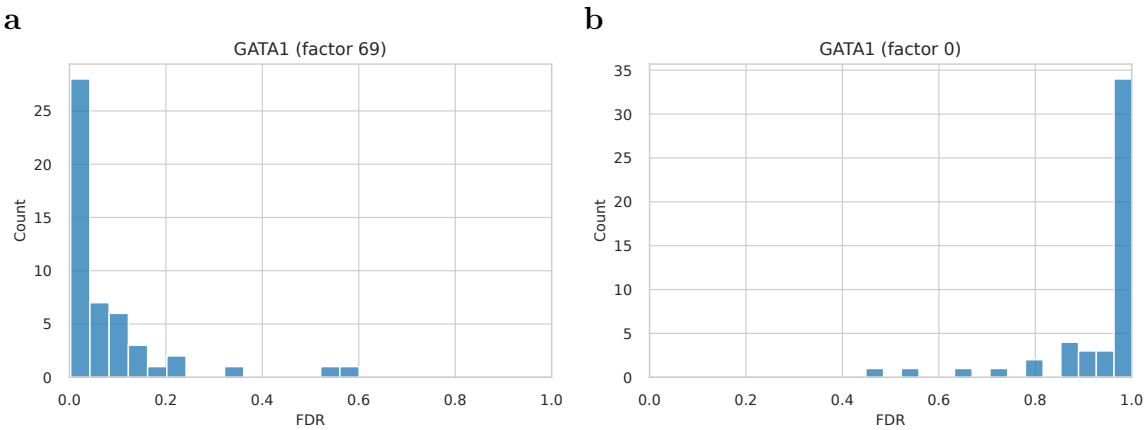


Fig. 5.15 Distribution of the GATA1 false discovery rates in factor 69 (FDR < 0.05 in 28/50 metagenes) and an arbitrary factor (enriched in 0/50 metagenes).

5.3 Discussion

Effective multi-tissue omics integration promises a system-wide view of human physiology, with potential to shed light on intra- and multi-tissue molecular phenomena. Such an approach challenges single-tissue and conventional integration techniques — often unable to model a variable number of tissues with sufficient statistical strength — necessitating the development of scalable, non-linear, and flexible methods. Here we developed HYFA (**Hy**pergraph **Fa**ctorisation), a parameter-efficient approach for joint multi-tissue and cell-type gene expression imputation that imposes strong inductive biases to learn entity-independent relational semantics and demonstrates excellent imputation capabilities.

We performed extensive benchmarks on data from the Genotype-Tissue Expression project (GTEx; [12]; v8 and v9), the most comprehensive human transcriptome resource available, and evaluated imputation performance over a broad collection of tissues and cell types. In addition to standard transcriptome imputation approaches, we compared our method with TEEBoT [21], a linear method that predicts target gene expression from the principal components of the reference expression. In the single-tissue reference scenario, HYFA and TEEBoT attained comparable imputation performance, outperforming standard methods. In the multi-tissue reference scenario, HYFA consistently outperformed TEEBoT and standard approaches in all target tissues, demonstrating HYFA's capabilities to borrow non-linear information across a variable number of tissues and exploit shared molecular patterns.

In addition to imputing tissue-level transcriptomics, we investigated the ability of HYFA to predict cell-type-level gene expression from multi-tissue bulk expression measurements. Through transfer learning, we trained HYFA to infer cell-type signatures from a cohort of single-nucleus RNA-seq [215] with matching GTEx-v8 donors. The inferred cell-type signatures exhibited a strong correlation with the ground truth despite the low sample size, indicating that HYFA's latent representations are rich and amenable to knowledge transfer. Strikingly, HYFA also recovered cell-type profiles from tissues that were never observed at transfer time, pointing to HYFA's ability to leverage gene expression programs underlying cell-type identity [267] even in tissues that were not considered in the original study [215]. HYFA may also be used to impute the expression of disease-related genes in a tissue of interest (Supplementary Information D.15).

In post-imputation analysis, we studied whether the imputed data improves eQTL discovery. We employed HYFA to impute the gene expression levels of every uncollected tissue in GTEx-v8, yielding a complete dataset, and performed eQTL mapping.

Compared to the original dataset, we observed a substantial gain in number of genes with detected eQTLs, with kidney cortex showing the largest gain. The increase was highest for tissues with low sample sizes, which are the ones expected to benefit the most from knowledge sharing across tissues. Notably, HYFA's detected eQTLs with their target eGenes could be replicated using independent, single-tissue transcriptome datasets that focus on *depth*, including the blood eQTLGen [245] and the brain frontal cortex PsychENCODE [246] datasets. Moreover, we found a significant enrichment for experimentally validated causal variants from the Massively Parallel Reporter Assay ([247]) dataset. Our results uncover a large number of previously undetected tissue-specific eQTLs and highlight the ability of HYFA to exploit shared regulatory information across tissues.

Lastly, HYFA can provide insights on coordinated gene regulation and expression mechanisms across tissues. We analysed to what extent tissues from the gastrointestinal system are informative of gene expression in brain tissues — an important question that may shed light on the biology of the brain-gut axis — and identified enriched biological processes and molecular functions. Through Gene Set Enrichment Analysis [85], we observed, among the HYFA-learned metagenes, a substantial amount of enriched pathways, transcription factors, and known regulators of biological processes, opening the door to biological interpretations. Future work might also seek to impose stronger inductive bias to ensure that metagenes are identifiable and robust to batch effects.

We believe that HYFA, as a versatile graph representation learning framework, provides a novel methodology for effective integration of large-scale multi-tissue biorepositories. The hypergraph factorisation framework is flexible (it supports k-uniform hypergraphs of arbitrary node types) and may find application beyond computational genomics.

Chapter 6

Neighbourhood-aware mapping of tissue architectures in spatial transcriptomics

Analysing the spatial organisation of cells within a tissue can shed light on fundamental biological processes, including intercellular communication [23] and organogenesis [24], and mechanisms of diseases like cancer, diabetes, and autoimmune disorders [25–27]. Spatial transcriptomics technologies have recently enabled gene expression profiling *in situ*, but they often lack single-cell resolution, impeding fine-grained characterisation of cellular heterogeneity and effective reconstruction of tissue architectures.

Computational approaches for cell-type deconvolution in spatial transcriptomics offer a scalable solution to these challenges. These strategies often identify resident cell types from the RNA sequencing of dissociated single cells, yielding cell-type-specific gene expression signatures, and then infer the cell-type composition of every profiled spot [269–271]. A cutting-edge method in this family is Cell2Location [96], a Bayesian deconvolution approach that captures cell-type relationships through a hierarchical model and handles technical sources of variation like differences in mRNA detection sensitivity. Despite numerous benefits, however, existing deconvolution approaches treat spots independently of each other.

In this chapter, we investigate whether incorporating spatio-relational information leads to improved cell-type mapping. Building on the observation that neighbouring spots often exhibit similar cell-type compositions (Figure 6.1), we extend Cell2Location

---

The research presented in this chapter has been conducted in collaboration with Paul Scherer (equal contribution), Nikola Simidjievski, Mateja Jamnik, and Pietro Liò

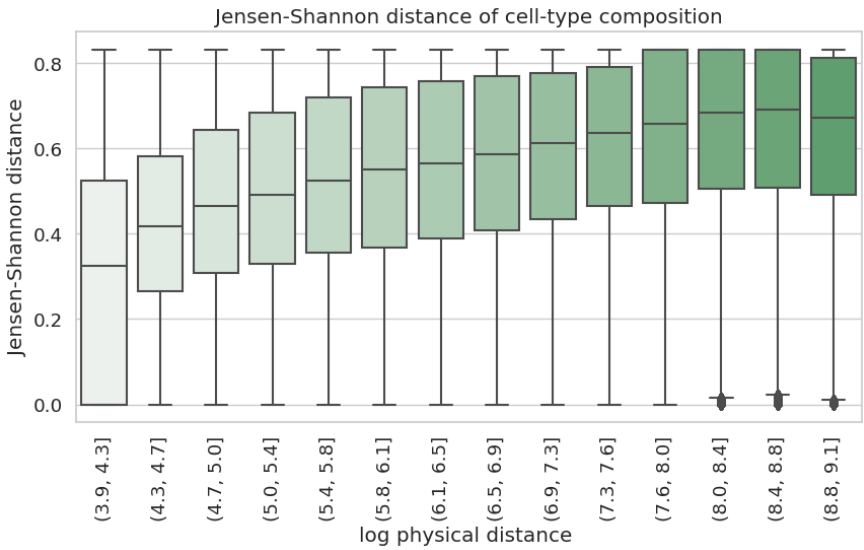


Fig. 6.1 Jensen-Shannon distance of cell-type proportions by spot distance in Xenium dataset (breast cancer, convolved spots of size 50µm). Closer spots tend to exhibit similar cell-type composition.

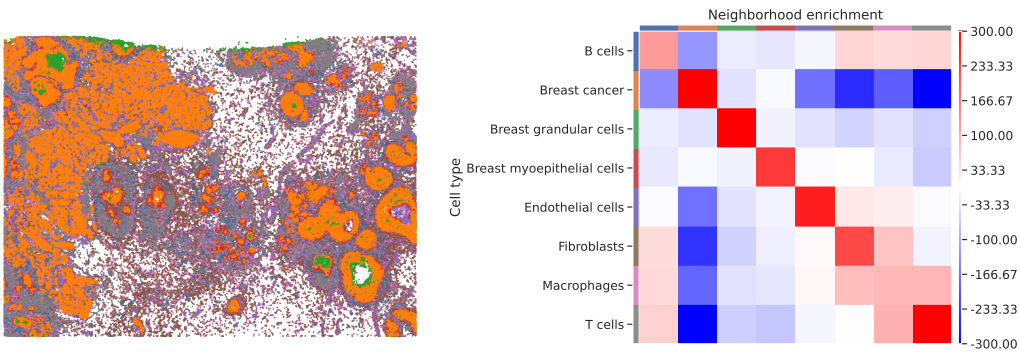


Fig. 6.2 Neighbourhood enrichment analysis on the Xenium dataset (breast cancer). The color legend is given by the y-axis of the neighbourhood heatmap. (*left*) Spatial transcriptomics data colored by cell-type. (*right*) Neighbourhood enrichment z-scores (red and blue indicate enrichment and depletion in the neighbourhood of nearest neighbours, respectively). Cells from the same cell-type tend to co-locate (e.g. breast cancer cells). Immune cells — including T cells, B cells, and macrophages — work in conjunction to modulate the anti-cancer immune response [268]. Utilising relational inductive biases could therefore enhance the effectiveness of spatial deconvolution models, thereby improving the characterisation of tumor microenvironments at different stages of cancer progression.

(C2L) to incorporate spatial inductive biases. Our approach, named GNN-C2L, propagates learnable messages on the proximity graph of spot transcripts, effectively leveraging the spatial relationships between spots and exploiting the co-location of cell-types (Figure 6.2). We conduct an extensive ablation study on synthetic and real spatial transcriptomics datasets and show improved deconvolution performance of GNN-C2L over spatial-agnostic variants. Altogether, our work leverages proximal inductive biases to facilitate an enhanced reconstruction of tissue architectures. Our code is publicly available at: `https://github.com/paulmorio/GNN-C2L`

6.1 Methodology

6.1.1 Problem formulation

Problem formulation Let $\boldsymbol{D} \in \mathbb{R}^{S \times G}$ denote a count matrix of RNA reads captured at S spots for G genes, using one or multiple batches (e.g. 10x Visium slides or Slide-seq pucks). Let $d_{s,g}$ be the entry of this matrix with the number of reads for gene g in spot s. Let $\boldsymbol{C} \in \mathbb{R}^{F \times G}$ denote a matrix of F reference cell-type signatures for the same set of G genes (e.g. these signatures can be obtained from dissociated single-cell RNA-seq). Denote by $c_{f,g}$ the expression of gene g in signature f. Given the count matrix $\boldsymbol{D}$ and cell-type signatures $\boldsymbol{C}$, our goal is to infer the cell-type composition $\boldsymbol{X} \in \mathbb{R}^{S \times F}$ of every spot.

6.1.2 Cell-type deconvolution with Cell2location

Cell2Location Our relational approach builds on Cell2Location [96], which models the per-spot read counts $\boldsymbol{D}$ as Negative Binomial (NB) distributed:

$$d_{s,g} \sim \mathrm{NB}(\mu_{s,g}, \alpha_{e,g}),$$

where $\alpha_{e,g}$ is an experiment- and gene-specific over-dispersion parameter and the unobserved expression rate $\mu_{s,g}$ is modelled as a linear function of the reference cell-type signatures $c_{f,g}$:

$$\mu_{s,g} = \left(m_g \cdot \sum_f w_{s,f} c_{f,g} + s_{e,g} \right) \cdot y_s,$$

where $w_{s,f}$ corresponds to the abundance of cell-type f at location s, m_g is a scaling parameter specific to gene g, $s_{e,g}$ is an experiment- and gene-specific additive shift,

and y_s is the detection sensitivity at spot s. Cell2location further places a hierarchical prior on $w_{s,f}$ to borrow statistical strength across groups of cell-types [96]. The priors on the model's parameters are described in full detail in the supplementary materials of [96].

6.1.3 Incorporating spatio-relational inductive biases

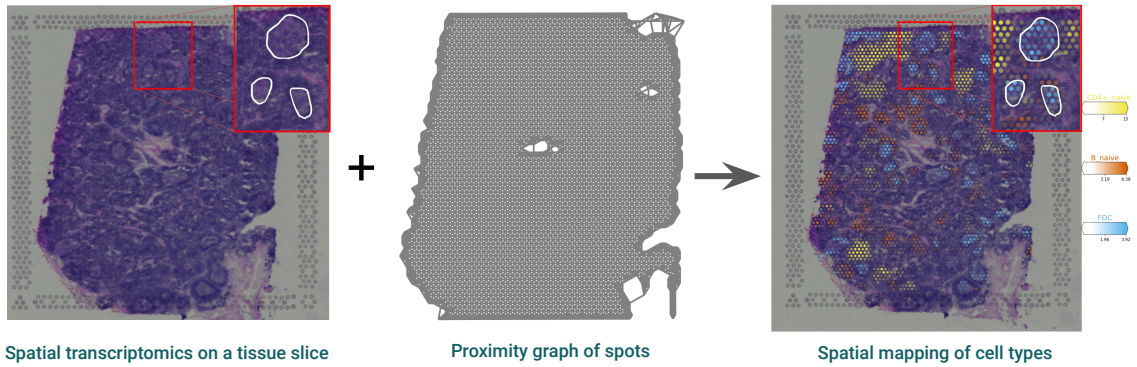


Fig. 6.3 Overview of the GNN-C2L framework. We develop a method that resolves the cell-type composition of every spot in a spatial transcriptomics dataset. In contrast to existing methods, GNN-C2L leverages spatial information through a proximity graph of spots. The right figure depicts the resolved cell-type abundances of three arbitrary cell-types on a Visium dataset [96] as predicted by our method. Figure credits: Paul Scherer, edited with permission.

GNN-C2L We propose a hierarchical model for cell-type composition inference that incorporates proximal relationships between spots. Let $\mathcal{N}(s)$ be the set of neighbour indices for spot s. This set of neighbours can be adapted to various spatial arrangements (e.g. hexagonal neighbourhoods for 10X Visium data) and k-hop neighbourhoods. To account for the neighbourhood information, we introduce a latent variable $\gamma_{s,f}$ representing the neighbour-aware cell-type abundances:

$$\begin{aligned}\gamma_{s,f} &\sim \text{Gamma}(\kappa_{s,f}, 1)\\ \boldsymbol{\kappa}_s &= \psi\Big(\boldsymbol{w}_s, \{\{\boldsymbol{w}_j \mid j \in \mathcal{N}(s)\}\}\Big),\end{aligned}$$

where the shape parameter $\kappa_{s,f}$ depends on the latent variables $\boldsymbol{w}_s$ and $\{\{\boldsymbol{w}_j \mid j \in \mathcal{N}(s)\}\}$ of spot s and its neighbours through a transformation $\psi(\cdot)$. Unlike Cell2Location, this effectively adds graphical dependencies between the neighbour-informed variables

$\gamma_{s,f}$ and the latent variables $w_{s,f}$. Importantly, computing $\gamma_{s,f}$ as a function of $\boldsymbol{w}_s$ allows capturing cell-type co-location patterns.

We then compute mean parameter $\mu_{s,g}$ of the Negative Binomial $\mathrm{NB}(\mu_{s,g}, \alpha_{e,g})$ likelihood using the neighbour-aware cell-type abundances $\gamma_{s,f}$:

$$\mu_{s,g} = \Big(m_g \cdot \sum_f \gamma_{s,f} c_{f,g} + s_{e,g}\Big) \cdot y_s$$

For all parameters, we utilise the validated hierarchical priors and hyperpriors of Cell2Location [96].

Incorporating spatial inductive biases The form of $\psi(\cdot)$ determines the inductive biases of the model. In this study, we construct a proximity graph of spatially localised spots, i.e. we consider physically adjacent spots, allowing for different spatial arrangements (e.g. hexagonal neighbourhoods for 10X Visium data) and k-hop neighbourhoods. We consider several graph neural network architectures for $\psi(\cdot)$, starting with simple graph convolutional network [272] to validate whether homophily (brought about by feature propagation) is a useful inductive bias, and introducing other GNN operators to allow for a more expressive use of the available spatio-relational data. We also consider a standard multi-layer perceptron as a baseline to assess whether performance changes can be attributed to similarly parametrised spatial-agnostic transformations. We next describe the alternatives for ψ in greater detail.

MLP-C2L As a spatial-agnostic control, we model $\psi(\cdot)$ with an MLP, i.e. $\boldsymbol{\kappa}_s = \mathrm{MLP}(\boldsymbol{w}_s)$, using a softplus activation function. This model does not utilise any spatial relationships between the spots and, alongside Cell2Location, serves as a control for our hypothesis.

SGC-C2L We construct a GNN-C2L variant using Simple Graph Convolutional (SGC) layers [272, 273]. Let $d_s = |\mathcal{N}(s)|$ be the node degree of spot s. A SGC layer computes the neighbour-aware features $\boldsymbol{\kappa}_s$ using a weighted average of the latent variables $\boldsymbol{w}_s$ in the local neighbourhood:

$$\begin{aligned}
\boldsymbol{\kappa}_s &= \mathrm{Linear}(\boldsymbol{h}_s) \\
\boldsymbol{h}_s &= \frac{1}{d_s+1}\boldsymbol{w}_s + \sum_{j \in \mathcal{N}(s)} \frac{1}{\sqrt{(d_s+1)(d_j+1)}}\boldsymbol{w}_j
\end{aligned}$$

The feature propagation mechanism biases the representations $\boldsymbol{\kappa}_s$ of neighbouring spots to become more similar to each other, using a degree-normalised adjacency matrix with self-loops. Thus, this simple MLP extension encourages homophilous

latent cell-type distributions. Optionally, we can apply an activation function after the linear transformation and stack several SGC layers to expand the receptive field.

GAT-C2L We increase the expressivity of $\psi(\cdot)$ by utilising graph attention networks, specifically GATv2 [221]. Unlike the constant, degree-dependant neighbouring contribution in the SGC-C2L model, the GATv2-C2L variant employs a learnable attention mechanism with increased control of contribution strengths, allowing to capture both homophilic and cell-type co-location patterns:

$$\boldsymbol{\kappa}_s = \alpha_{s,s}\phi(\boldsymbol{w}_s) + \sum_{j\in\mathcal{N}(s)} \alpha_{s,j}\phi(\boldsymbol{w}_j),$$

where ϕ is an MLP with a softplus activation function. We define the attention coefficient $\alpha_{s,j}$ as:

$$\alpha_{s,j} = \frac{\exp e(\boldsymbol{w}_s, \boldsymbol{w}_j)}{\sum_{k\in\mathcal{N}(s)\cup\{s\}} \exp e(\boldsymbol{w}_s, \boldsymbol{w}_k)}$$
$$e(\boldsymbol{w}_s, \boldsymbol{w}_j) = \mathbf{a}^T \mathrm{LeakyReLU}(\boldsymbol{W}[\boldsymbol{w}_s || \boldsymbol{w}_j]),$$

where $||$ is the concatenation operation and $\mathbf{a}$ and $\boldsymbol{W}$ are learnable parameters shared across spots, allowing the neural network to mix signals over the different cell types.

Training and inference We approximate the model parameters through variational inference. For every latent variable, we use a univariate normal distribution to approximate the posterior and utilise a softplus activation to ensure a positivity. Minimisation of the ELBO jointly trains the parameters of the model (and the incorporated GNNs) as well as the variational distribution. After optimisation, we estimate the cell-type abundances of every spot s by averaging $\gamma_{s,f}$ over 1000 samples of the variational distribution.

6.1.4 Experimental setup

We study whether incorporating spatial relationships via graph neural networks leads to enhanced cell-type mapping.

Datasets To quantitatively benchmark the baselines, we utilised a synthetic dataset introduced in Cell2Location [96] for which we knew the true cell-type abundances of each spot. The construction of this dataset is detailed extensively in [96]. Moreover, we evaluated the methods using two real datasets, MPOA [274] and Xenium (breast

cancer) [275], that have single-cell resolution (yet fewer genes are profiled). To simulate real spots, we divided the tissues into squared spots of size 100μm and summed the expression of all cells within every square. For the Xenium and MPOA datasets, we constructed the cell-type signatures by averaging the read counts of all cells from every given cell-type.

Hyperparameter settings We used the same hyperparameters for every baseline where applicable. We set the hidden dimensions of each layer to 64 and used a single GNN layer (i.e. 1-hop receptive field). We conducted an ablation study using more graph layers in Supplementary Information E. We minimised the variational lower bound using Adam [187] with learning rate of 0.001 for 25,000 epochs in all datasets.

Evaluation metrics For all datasets we assessed performance using the average Pearson R correlation, Jensen-Shannon Divergence (JSD), and the area under precision-recall curve (AUPRC) (macro-averaged over cell-types) between the ground-truth and inferred cell-type proportions. We computed Pearson R over the flat ground-truth and inferred cell-type proportions. We calculated the Jensen-Shannon Divergence between the per-spot ground-truth and inferred cell-type proportions. We binarised the true cell abundance matrix to show which cell types were present in which locations, and then used the inferred cell-type proportions to compute the AUPRC.

6.2 Results and discussion

We benchmark spatial-agnostic (Cell2location, GNN-C2L MLP) and spatial-aware GNN-C2L (SGC, GAT) baselines on simulated and semi-simulated (MPOA and Xenium) spatial transcriptomics data (Table 6.1).

Results on simulated dataset We studied deconvolution performance on the synthetic data over: 1) **ALL**: all cell types, 2) ubiquitious high cell abundance (**UHCA**): 3 high-abundance cell types spatially distributed in uniform manner across the tissue, 3) ubiquitious low cell abundance (**ULCA**): 5 low-abundance cell types spatially distributed in uniform manner across the tissue, 4) regional high cell abundance (**RHCA**): 9 cell types with local distribution patterns, i.e. cell types cluster in specific locations with high abundance and exhibit 0 abundance elsewhere, 5) regional low cell abundance (**RLCA**): 32 low-abundance cell types that have local distribution patterns.

Table 6.1 Average Pearson R, Avg. Jensen Shannon divergence (JSD), and AUPRC scores and standard deviation of 5 seeded runs of each model over all spots. For the synthetic dataset, scores for subcategories of cell types exhibiting distinct cell abundance patterns are also provided. Bold numbers indicate best-performing method for each category of cell types being evaluated for each metric. Overall, the GNN-C2L spatial-aware variants attained equal or superior deconvolution scores than spatial-agnostic baselines. Table credit: Paul Scherer.

	Simulated					Semi-simulated		
Method	ALL	UHCA	ULCA	RHCA	RLCA	MPOA	Xenium	Metric
Cell2location	0.683 ± 0.002	0.882 ± 0.001	0.519 ± 0.007	0.836 ± 0.004	0.422 ± 0.003	0.929 ± 0.001	0.928 ± 0.002	R
GNN-C2L (MLP)	0.672 ± 0.024	0.866 ± 0.008	0.661 ± 0.021	0.865 ± 0.007	0.404 ± 0.040	0.920 ± 0.005	**0.929 ± 0.000**	
GNN-C2L (SGC)	0.699 ± 0.023	0.876 ± 0.008	**0.708 ± 0.020**	0.883 ± 0.006	0.439 ± 0.041	0.936 ± 0.001	0.928 ± 0.000	
GNN-C2L (GAT)	**0.737 ± 0.013**	**0.885 ± 0.018**	0.695 ± 0.032	**0.888 ± 0.004**	**0.492 ± 0.032**	**0.936 ± 0.001**	0.928 ± 0.000	
Cell2location	0.468 ± 0.001	**0.202 ± 0.002**	0.496 ± 0.001	0.421 ± 0.002	0.509 ± 0.001	0.204 ± 0.001	0.213 ± 0.005	Avg. JSD
GNN-C2L (MLP)	0.457 ± 0.006	0.230 ± 0.012	0.473 ± 0.007	0.387 ± 0.006	0.503 ± 0.009	0.199 ± 0.004	0.211 ± 0.001	
GNN-C2L (SGC)	0.446 ± 0.006	0.224 ± 0.011	0.460 ± 0.007	**0.368 ± 0.005**	0.493 ± 0.009	0.189 ± 0.001	**0.211 ± 0.001**	
GNN-C2L (GAT)	**0.435 ± 0.003**	0.209 ± 0.021	**0.458 ± 0.014**	0.369 ± 0.001	**0.482 ± 0.006**	**0.188 ± 0.001**	0.212 ± 0.000	
Cell2location	0.591 ± 0.003	0.932 ± 0.006	0.477 ± 0.005	0.783 ± 0.003	0.591 ± 0.003	**0.956 ± 0.001**	0.873 ± 0.003	AUPRC
GNN-C2L (MLP)	0.675 ± 0.002	0.963 ± 0.006	0.590 ± 0.004	0.804 ± 0.004	0.675 ± 0.002	0.951 ± 0.001	0.883 ± 0.001	
GNN-C2L (SGC)	0.719 ± 0.002	0.977 ± 0.004	0.646 ± 0.006	**0.861 ± 0.001**	0.719 ± 0.002	0.955 ± 0.000	0.884 ± 0.000	
GNN-C2L (GAT)	**0.722 ± 0.002**	**0.978 ± 0.004**	**0.664 ± 0.004**	0.858 ± 0.003	**0.722 ± 0.002**	0.952 ± 0.001	**0.884 ± 0.000**	

Overall, GNN-C2L consistently outperformed the spatial-agnostic baselines on the synthetic data (Table 6.1). We observed a marked increase in performance through the utilisation of proximal relations across different metrics and subtasks. Spatial-aware baselines achieved the best scores in 13 out of 15 cases, especially for cell types with low cell abundance (ULCA and RLCA). The performance difference was particularly apparent from the overall scores of the MLP variant of GNN-C2L (ALL R: 0.672 ± 0.024, JSD: 0.457 ± 0.006, AUPRC: 0.675 ± 0.002) and GNN-C2L SGC (ALL R: 0.699 ± 0.023, JSD: 0.446 ± 0.006, AUPRC: 0.719 ± 0.002) — both baselines utilised the same amount of learnable parameters, yet only GNN-C2L (SGC) propagates information across spots. It is also worth noting that using additional parameters may result in degraded performance, i.e. compared to Cell2Location (ALL $R : 0.683 \pm 0.002$), GNN-C2L (MLP) attained reduced Pearson R correlation and increased variance (ALL $R : 0.672 \pm 0.024$). Altogether, our results highlight the superior ability of GNN-C2L to perform cell-type deconvolution.

Results on semi-simulated datasets In performance comparison on the semi-simulated datasets (MPOA and Xenium), the spatial-aware GNN-C2L variants achieved equal or better deconvolution performance than the spatial-agnostic baselines (Table 6.1). On MPOA, all baselines performed well — it is worth noting that this is a considerably smaller dataset with larger spot sizes (per-spot average of 18 cells) compared to the synthetic ($\sim$9 cells per spot) and Xenium ($\sim$10 cells per spot) datasets.

This may have an effect on the specificity of the transcript readings as well as the usefulness of local information considering the size of micro-architectures in the tissue. We observed that GAT-C2L had the best scores in 2 out of 3 metrics (R: 0.492 ± 0.032, JSD: 0.188 ± 0.001), while Cell2Location was superior in terms of AUPRC (0.956 ± 0.001). In the Xenium dataset, all baselines attained comparable results (e.g. Cell2location R: 0.928 ± 0.000, GAT R: 0.928 ± 0.000; Cell2location AUPRC: 0.873 ± 0.003, MLP AUPRC: 0.883 ± 0.001, SGC AUPRC: 0.884 ± 0.000).

Conclusion In this chapter, we introduced an approach for spatial cell-type deconvolution. Our method (GNN-C2L), builds on Cell2Location [96] to predict the per-spot cell-type composition in spatial transcriptomic datasets lacking single-cell resolution. In contrast to Cell2Location (spatial-agnostic), GNN-C2L incorporates inductive biases to predict neighbourhood-aware cell-type abundances at every spot, which enables capturing homophilic and cell-type co-location patterns. In performance comparison, GNN-C2L achieved comparable or improved deconvolution performance on simulated and semi-simulated datasets with ground-truth information. Collectively, our results suggest that spatial deconvolution can benefit from spatio-relational inductive biases, with potential for an enhanced reconstruction of tissue architectures.

Broader impact Characterising molecular information in the spatial domain can greatly enhance our understanding about cell-cell communication and coordination to attain high-level functions within a tissue (e.g. brain function [276]) and fight diseases (e.g. the role of immune cells in cancer [268]). As spatial technologies continue to develop, computational approaches for modelling spatial transcriptomics will likely find application in clinical diagnosis and personalised treatment of diseases [277]. From a modelling standpoint, leveraging proximity networks of cells, as done in this chapter through spatio-relational inductive biases, might allow us to uncover spatially sensitive biomarkers and detect disease-specific signaling events [278], potentially leading to improved diagnosis, prognosis, and treatments.

Chapter 7

Conclusions

In this thesis, we have developed computational methods for modelling gene expression data, focusing on its tissue-specificity and enabling several downstream applications. These include the generation of transcriptomic data in-silico, gene expression imputation from a subset of measured genes and across multiple collected tissues, and characterisation of tissue architectures using spatial transcriptomics. This chapter summarises the main contributions of the dissertation and highlights further avenues for future work in this domain.

7.1 Summary of contributions

The main contributions of the dissertation are:

- In **Chapter 3**, we developed a generative model of transcriptomic data based on Wasserstein generative adversarial networks with gradient penalty (WGAN-GP) [18]. We studied the degree of realism of the in-silico generated data in two transcriptomic datasets, including an *Escherichia coli* (*E. coli*) dataset (an organism for which regulatory interactions are well-characterised) and a multi-tissue expression dataset consisting of healthy and cancer samples. We evaluated several key properties of gene expression (e.g. clustering patterns and regulatory interactions) and found that, in contrast to existing simulators of gene expression, WGAN-GP faithfully preserved these patterns. We further utilised this method to generate tissue-specific gene expression data of the synthetic individuals in two conditions (healthy and cancer) and recapitulated several cancer biomarkers through a sensitivity analysis.

- In **Chapter 4**, we introduced two computational models for the imputation of gene expression within a single tissue, studying whether the full transcriptome can be

recovered from smaller subsets of genes with minimal reconstruction error. The first method, pseudo-mask imputation (PMI), is a self-supervised technique that dynamically imputes the expression of a subset of pseudo-missing genes as a function of the remaining observed genes. The second model, GAIN-GTEx, is based on generative adversarial imputation networks [184]. We benchmarked performance in two case studies (protein-coding genes and genes from the Alzheimer's disease pathway) and two imputation scenarios (inductive and in-place imputation) across a broad collection of tissues. We showed that the proposed approaches compared favourably to standard and state-of-the-art imputation techniques, both in terms of imputation performance and runtime. We also evaluated the imputation capabilities on transcriptomic data from 3 independent cancer datasets and observed strong generalisation across varying levels of missingness.

- In **Chapter 5**, we presented Hypergraph Factorisation (HYFA), a parameter-efficient graph representation learning approach for multi-tissue gene expression imputation. HYFA imputes tissue-specific gene expression via a specialised graph neural network operating on a hypergraph of individuals, metagenes, and tissues. HYFA is genotype-agnostic, supports a variable number of collected tissues per individual, and imposes strong inductive biases to leverage the shared regulatory architecture of tissues. In performance comparison, HYFA achieved superior performance over existing transcriptome imputation methods, especially when multiple reference tissues were available. Through transfer learning on a paired single-nucleus RNA-seq (snRNA-seq) dataset, we further showed that HYFA can resolve cell-type signatures from bulk gene expression, highlighting the method's ability to leverage gene expression programs underlying cell-type identity, even in tissues that were never observed in the training set. Using Gene Set Enrichment Analysis, we found that the metagenes learned by HYFA capture information about known biological pathways. Notably, the HYFA-imputed dataset generated a large catalog of new tissue-specific expression Quantitative Trait Loci (eQTLs). HYFA's detected eQTLs could also be replicated in independent datasets and were enriched for experimentally-validated causal variants.

- In **Chapter 6**, we studied the spatial deconvolution problem. Given a spatial transcriptomic dataset where gene expression is profiled *in-situ* but not at single-cell resolution, the goal is to infer cell-type abundances at each spatial location of the tissue. Several techniques have been proposed to address this problem [269–271, 96], but existing approaches treat neighbouring spots independently

of each other. To address this limitation, we extended the Cell2location [96] methodology by incorporating spatio-relational inductive biases that allow estimation of cell-type abundances in a neighbour-aware manner. Our approach, named GNN-C2L, propagates learnable messages on the proximity graph of spots, effectively leveraging the spatial relationships between spots and exploiting the co-location of cell-types. We conducted an extensive ablation study on synthetic and real spatial transcriptomics datasets and showed improved deconvolution performance of GNN-C2L over spatial-agnostic variants. We believe that accounting for spatial inductive biases may facilitate an enhanced reconstruction of tissue architectures.

7.2 Future work

The rapid technical advances and declining costs of sequencing technologies will generate an unprecedented amount of omics data across multiple tissues and cell-types, accompanied by novel methodological problems and opportunities. Some of the broad methodological challenges include integrating heterogeneous omics data across modalities [3, 4], tissues [2, 5], experimental settings [6], and species [7]; dealing with high-dimensional data in combination with a scarce number of labelled samples [8]; imputing missing or unreliable values [2]; identifying causal relationships rather than mere statistical associations [279]; generalising under distribution shifts [280]; ensuring algorithmic fairness [281]; validating and benchmarking computational tools in a systematic way [1]; and interpreting deep learning models [9]. In particular, some promising avenues for further research and innovations are:

- **Transcriptome-wide association studies**. Genome-wide association studies (GWAS) have identified thousands of genetic variants associated with complex traits, including mental disorders like Alzheimer's disease [282] and physical diseases like coronary artery disease [283]. GWAS can predict disease susceptibility based on rare mutations and may soon be used in clinical settings [284, 285], but inferring causal variants is complicated from GWAS studies alone [279, 285]. Transcriptome-wide association studies (TWAS) aim to narrow down the large pool of genomic variants identified in GWAS by considering the individuals' transcriptomes, which constitute the intermediate step between their genetic information and complex traits. TWAS first trains a model that predicts expression from the genotype on a reference dataset (e.g. GTEx), then applies the model to individuals of the GWAS cohort, and finally identifies associations between the predicted expression and the phenotype [279]. However, one of the main issues of TWAS studies is tissue

bias — where the TWAS tissue is not mechanistically related to the complex trait [279], e.g. due to a low-sample size of the mechanistically related tissue. In this case, multi-tissue gene expression imputation approaches such as HYFA [5], which leverage the shared regulatory architecture of tissues, may be used to impute the uncollected samples in the tissue of interest.

- **Predicting the effect of genetic perturbations**. Reasoning about causal relationships requires going beyond traditional statistics. In the standard statistical framework, the joint probability distribution from which the data is drawn is assumed to be *static*, that is, the conditions under which the data is generated do not vary across observations. While this allows us to describe associations between genes, it fails to capture *dynamic* properties of the world, such as how the behavior of a particular gene changes when an unexpected agent intervenes another gene. This latter case concerns a causal relationship that cannot be described merely as a conditional probability distribution. The difference between the *static* and *dynamic* scenario corresponds to the *basic distinction of causality* [286]. Recent advances in gene editing techniques (e.g. CRISPR [287]) have enabled the generation of interventional transcriptomic data [288] with broad applications. Methods that can predict the transcriptional effect of genetic perturbations may play a pivotal role in the elucidation of tissue- and cell-type-specific gene regulatory interactions, the discovery of disease mechanisms, and the development of personalised drugs [289].

- **Personalised medicine and digital twins in healthcare**. Our ability to measure the molecular characteristics of an individual opens the door to promising applications in personalised medicine [290], that is, the diagnosis, prevention, and treatment of diseases in a way that is optimally tailored to each individual. As multi-omic technologies become cheaper and more scalable, collecting longitudinal omics information will allow monitoring of the physiological state of individuals [290] and characterising dysregulated processes [291]. Methods that integrate molecular and physiological information may give rise to the first generation of *digital twins* in healthcare, providing a system-wide view of human physiology across multiple organs [5] and scales [31]. This may allow experimenting with multiple personalised therapies and predicting disease trajectories in a minimally invasive and cost-effective way [31]. Alternatively, integration of omics datasets with large perturbational datasets [292] could enable personalised treatment recommendations based on the individuals' molecular characteristics.

- **Single-cell data integration and foundation methods**. Global efforts such as the human cell atlas [49] and the mouse cell atlas [293] have created comprehensive maps of cells under different conditions and in multiple tissues and organisms. These efforts can increase our understanding of cell biology [7] and life's most fundamental principles [49], but demand novel methodological advances. Single-cell data is known to be substantially noisy and susceptible to *batch effects*, and technical sources of variation may act as confounders for the true biological signal, limiting our ability to identify population-level differences. Furthermore, independent studies might profile different sets of genes in different cell populations, which complicates downstream analyses. Thus, flexible methods that can integrate single-cell data across different gene sets [294], experimental settings [6], omics modalities [295], and species [7] will facilitate the joint analysis of millions of cells, with potential to characterise biological processes [49], unravel regulatory networks across genes [294] and omics layers [295], discover novel cell-types [296], and accelerate the discovery of therapeutic targets [294].

References

[1] David Lähnemann, Johannes Köster, Ewa Szczurek, Davis J McCarthy, Stephanie C Hicks, Mark D Robinson, Catalina A Vallejos, Kieran R Campbell, Niko Beerenwinkel, Ahmed Mahfouz, et al. Eleven grand challenges in single-cell data science. *Genome biology*, 21(1):1–35, 2020.

[2] Ramon Viñas, Tiago Azevedo, Eric R. Gamazon, and Pietro Liò. Deep learning enables fast and accurate imputation of gene expression. *Frontiers in Genetics*, 12, 2021.

[3] David Gomez-Cabrero, Imad Abugessaisa, Dieter Maier, Andrew Teschendorff, Matthias Merkenschlager, Andreas Gisel, Esteban Ballestar, Erik Bongcam-Rudloff, Ana Conesa, and Jesper Tegnér. Data integration in the era of omics: current and future challenges. *BMC systems biology*, 8(2):1–10, 2014.

[4] Junyue Cao, Darren A Cusanovich, Vijay Ramani, Delasa Aghamirzaie, Hannah A Pliner, Andrew J Hill, Riza M Daza, Jose L McFaline-Figueroa, Jonathan S Packer, Lena Christiansen, et al. Joint profiling of chromatin accessibility and gene expression in thousands of single cells. *Science*, 361(6409):1380–1385, 2018.

[5] **Ramon Viñas**, Chaitanya Joshi, Dobrik Georgiev, Phillip Lin, Bianca Dumitrascu, Eric R. Gamazon, and Pietro Liò. Hypergraph factorisation for multi-tissue gene expression imputation. *Nature Machine Intelligence*, 2023.

[6] Brian Hie, Bryan Bryson, and Bonnie Berger. Efficient integration of heterogeneous single-cell transcriptomes using scanorama. *Nature biotechnology*, 37(6):685–691, 2019.

[7] Yanay Rosen, Maria Brbić, Yusuf Roohani, Kyle Swanson, Ziang Li, and Jure Leskovec. Towards universal cell embeddings: Integrating single-cell rna-seq datasets across species with saturn. *Biorxiv: the Preprint Server for Biology*, 2023.

[8] Robert Clarke, Habtom W Ressom, Antai Wang, Jianhua Xuan, Minetta C Liu, Edmund A Gehan, and Yue Wang. The properties of high-dimensional data spaces: implications for exploring gene and protein expression data. *Nature reviews cancer*, 8(1):37–49, 2008.

[9] Christina B Azodi, Jiliang Tang, and Shin-Han Shiu. Opening the black box: interpretable machine learning for geneticists. *Trends in genetics*, 36(6):442–455, 2020.

[10] François Aguet and Kristin G Ardlie. Tissue specificity of gene expression. *Current Genetic Medicine Reports*, 4:163–169, 2016.

[11] Abhijeet Rajendra Sonawane, John Platig, Maud Fagny, Cho-Yi Chen, Joseph Nathaniel Paulson, Camila Miranda Lopes-Ramos, Dawn Lisa DeMeo, John Quackenbush, Kimberly Glass, and Marieke Lydia Kuijjer. Understanding tissue-specific gene regulation. *Cell reports*, 21(4):1077–1088, 2017.

[12] GTEx Consortium. The gtex consortium atlas of genetic regulatory effects across human tissues. *Science*, 369(6509):1318–1330, 2020.

[13] Alice Lacan, Michèle Sebag, and Blaise Hanczar. GAN-based data augmentation for transcriptomics: survey and comparative assessment. *Bioinformatics*, 39:i111–i120, 06 2023.

[14] Ramon Viñas, Helena Andrés-Terré, Pietro Liò, and Kevin Bryson. Adversarial generation of gene expression data. *Bioinformatics*, 01 2021. btab035.

[15] Tolou Shadbahr, Michael Roberts, Jan Stanczuk, Julian Gilbey, Philip Teare, Sören Dittmer, Matthew Thorpe, **Ramon Viñas**, Evis Sala, Pietro Liò, et al. Classification of datasets with imputed missing values: Does imputation quality matter? *Nature Communications Medicine*, 2023.

[16] **Ramon Viñas***, Paul Scherer*, Nikola Simidjievski, Mateja Jamnik, and Pietro Liò. Spatio-relational inductive biases in spatial cell-type deconvolution. *2023 ICML Workshop on Computational Biology*, 2023.

[17] Robert Maier, Ralf Zimmer, and Robert Küffner. A Turing test for artificial expression data. *Bioinformatics*, 29(20):2603–2609, 2013.

[18] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of Wasserstein GANs. *CoRR*, abs/1704.00028, 2017.

[19] Bin Zhang and Steve Horvath. A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, 4(1), 2005.

[20] Ana Conesa, Pedro Madrigal, Sonia Tarazona, David Gomez-Cabrero, Alejandra Cervera, Andrew McPherson, Michał Wojciech Szcześniak, Daniel J Gaffney, Laura L Elo, Xuegong Zhang, et al. A survey of best practices for rna-seq data analysis. *Genome biology*, 17(1):13, 2016.

[21] Mahashweta Basu, Kun Wang, Eytan Ruppin, and Sridhar Hannenhalli. Predicting tissue-specific gene expression from whole blood transcriptome. *Science Advances*, 7(14):eabd6991, 2021.

[22] Kazumasa Kanemaru, James Cranley, Daniele Muraro, Antonio MA Miranda, Siew Yen Ho, Anna Wilbrey-Clark, Jan Patrick Pett, Krzysztof Polanski, Laura Richardson, Monika Litvinukova, et al. Spatially resolved multiomics of human cardiac niches. *Nature*, pages 1–10, 2023.

[23] David S. Fischer, Anna C. Schaar, and Fabian J. Theis. Modeling intercellular communication in tissues using spatial graphs of cells. *Nature Biotechnology*, 41(3):332–336, Mar 2023.

[24] T Lohoff, S Ghazanfar, A Missarova, N Koulena, N Pierson, JA Griffiths, ES Bardot, C-HL Eng, RCV Tyser, R Argelaguet, et al. Integration of spatial and single-cell transcriptomic data elucidates mouse organogenesis. *Nature biotechnology*, 40(1):74–85, 2022.

[25] Giovanni Solinas, Cristian Vilcu, Jaap G. Neels, Gautam K. Bandyopadhyay, Jun-Li Luo, Willscott Naugler, Sergei Grivennikov, Anthony Wynshaw-Boris, Miriam Scadeng, Jerrold M. Olefsky, and Michael Karin. Jnk1 in hematopoietically derived cells contributes to diet-induced inflammation and insulin resistance without affecting obesity. *Cell Metabolism*, 6(5):386–397, 2007.

[26] Kepeng Wang, Sergei I Grivennikov, and Michael Karin. Implications of anti-cytokine therapy in colorectal cancer and autoimmune diseases. *Annals of the Rheumatic Diseases*, 72(suppl 2):ii100–ii103, 2013.

[27] Spiros A Vlahopoulos, Osman Cen, Nina Hengen, James Agan, Maria Moschovi, Elena Critselis, Maria Adamaki, Flora Bacopoulou, John A Copland, Istvan Boldogh, Michael Karin, and George P Chrousos. Dynamic aberrant NF-κB spurs tumorigenesis: a new model encompassing the microenvironment. *Cytokine Growth Factor Rev.*, 26(4):389–403, August 2015.

[28] **Ramon Viñas**, Helena Andrés-Terré, Pietro Liò, and Kevin Bryson. Adversarial generation of gene expression data. *Bioinformatics*, 01 2021.

[29] **Ramon Viñas**, Tiago Azevedo, Eric R. Gamazon, and Pietro Liò. Deep learning enables fast and accurate imputation of gene expression. *Frontiers in Genetics*, 12:489, 2021.

[30] Paris DL Flood, **Ramon Viñas**, and Pietro Liò. Investigating estimated kolmogorov complexity as a means of regularization for link prediction. *2020 NeurIPS workshop on Causality*, 2020.

[31] Pietro Barbiero*, **Ramon Viñas***, and Pietro Liò. Graph representation forecasting of patient's medical conditions: Toward a digital twin. *Frontiers in genetics*, 12:652907, 2021.

[32] Viola Fanfani, **Ramon Viñas**, Pietro Liò, and Giovanni Stracquadanio. Discovering cancer driver genes and pathways using stochastic block model graph neural networks. *bioRxiv*, 2021.

[33] James King, **Ramon Viñas**, Alexander Campbell, and Pietro Liò. An investigation of pre-upsampling generative modelling and generative adversarial networks in audio super resolution. *arXiv preprint arXiv:2109.14994*, 2021.

[34] Paul Scherer, Maja Trebacz, Nikola Simidjievski, **Ramon Viñas**, Zohrer Shams, Helena Andres Terre, Mateja Jamnik, and Pietro Liò. Unsupervised construction of computational graphs for gene expression data with explicit structural inductive biases. *Bioinformatics*, 38(5):1320–1327, 2022.

[35] Arian Jamasb, **Ramon Viñas**, Eric Ma, Yuanqi Du, Charles Harris, Kexin Huang, Dominic Hall, Pietro Lió, and Tom Blundell. Graphein - a Python library for geometric deep learning and network analysis on biomolecular structures and interaction networks. In *Advances in Neural Information Processing Systems*, 2022.

[36] Ben Day*, **Ramon Viñas***, Nikola Simidjievski, and Pietro Liò. Attentional meta-learners for few-shot polythetic classification. In *International Conference on Machine Learning*, pages 4867–4889. PMLR, 2022.

[37] Han-Bo Li, **Ramon Viñas**, and Pietro Liò. Improving classification and data imputation for single-cell transcriptomics with graph neural networks. In *NeurIPS 2022 AI for Science: Progress and Promises and NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*, 2022.

[38] Dobrik Georgiev*, **Ramon Viñas***, Sam Considine, Bianca Dumitrascu, and Pietro Liò. NARTI: Neural Algorithmic Reasoning for Trajectory Inference. *2023 ICML Workshop on Computational Biology*, 2023.

[39] Chaitanya K. Joshi, Arian R. Jamasb, **Ramon Viñas**, Charles Harris, Simon Mathis, and Pietro Liò. Multi-state RNA design with geometric multi-graph neural networks. *2023 ICML Workshop on Computational Biology*, 2023.

[40] Khan Academy. Overview: Eukaryotic gene regulation, Accessed May 9, 2023.

[41] S. Clancy and W. Brown. Translation: Dna to mrna to protein, 2008.

[42] Mirjana Efremova and Sarah A. Teichmann. Computational methods for single-cell omics across modalities. *Nature Methods*, 17:14–17, 2020.

[43] Yehudit Hasin, Marcus Seldin, and Aldons Lusis. Multi-omics approaches to disease. *Genome biology*, 18(1):1–15, 2017.

[44] Sam Behjati and Patrick S Tarpey. What is next generation sequencing? *Archives of Disease in Childhood-Education and Practice*, 98(6):236–238, 2013.

[45] Shikhar Sharma, Theresa K Kelly, and Peter A Jones. Epigenetics in cancer. *Carcinogenesis*, 31(1):27–36, 2010.

[46] Roy Lardenoije, Artemis Iatrou, Gunter Kenis, Konstantinos Kompotis, Harry WM Steinbusch, Diego Mastroeni, Paul Coleman, Cynthia A Lemere, Patrick R Hof, Daniel LA van den Hove, et al. The epigenetics of aging and neurodegeneration. *Progress in neurobiology*, 131:21–64, 2015.

[47] Patrik L Ståhl, Fredrik Salmén, Sanja Vickovic, Anna Lundmark, José Fernández Navarro, Jens Magnusson, Stefania Giacomello, Michaela Asp, Jakub O Westholm, Mikael Huss, et al. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science*, 353(6294):78–82, 2016.

[48] GTEx Consortium et al. The genotype-tissue expression (gtex) pilot analysis: multitissue gene regulation in humans. *Science*, 348(6235):648–660, 2015.

[49] Aviv Regev, Sarah A Teichmann, Eric S Lander, Ido Amit, Christophe Benoist, Ewan Birney, Bernd Bodenmiller, Peter Campbell, Piero Carninci, Menna Clatworthy, et al. The human cell atlas. *elife*, 6:e27041, 2017.

[50] Marlon Stoeckius, Christoph Hafemeister, William Stephenson, Brian Houck-Loomis, Pratip K Chattopadhyay, Harold Swerdlow, Rahul Satija, and Peter Smibert. Simultaneous epitope and transcriptome measurement in single cells. *Nature methods*, 14(9):865–868, 2017.

[51] Zhong Wang, Mark Gerstein, and Michael Snyder. Rna-seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, 10(1):57–63, 2009.

[52] Evan Z Macosko, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman, Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214, 2015.

[53] Matthias Meyer and Martin Kircher. Illumina sequencing library preparation for highly multiplexed target capture and sequencing. *Cold Spring Harbor Protocols*, 2010(6):pdb–prot5448, 2010.

[54] Anthony Rhoads and Kin Fai Au. Pacbio sequencing and its applications. *Genomics, proteomics & bioinformatics*, 13(5):278–289, 2015.

[55] Sara Goodwin, John D McPherson, and W Richard McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333–351, 2016.

[56] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by rna-seq. *Nature methods*, 5(7):621–628, 2008.

[57] Kasper D Hansen, Rafael A Irizarry, and Zhijin Wu. Removing technical variability in rna-seq data using conditional quantile normalization. *Biostatistics*, 13(2):204–216, 2012.

[58] Davide Risso, John Ngai, Terence P Speed, and Sandrine Dudoit. Normalization of rna-seq data using factor analysis of control genes or samples. *Nature biotechnology*, 32(9):896–902, 2014.

[59] Mark D Robinson and Alicia Oshlack. A scaling normalization method for differential expression analysis of rna-seq data. *Genome biology*, 11(3):1–9, 2010.

[60] W Evan Johnson, Cheng Li, and Ariel Rabinovic. Adjusting batch effects in microarray expression data using empirical bayes methods. *Biostatistics*, 8(1):118–127, 2007.

[61] Laleh Haghverdi, Aaron TL Lun, Michael D Morgan, and John C Marioni. Batch effects in single-cell rna-sequencing data are corrected by matching mutual nearest neighbors. *Nature biotechnology*, 36(5):421–427, 2018.

[62] Yalchin Oytam, Fariborz Sobhanmanesh, Konsta Duesing, Joshua C Bowden, Megan Osmond-McLeod, and Jason Ross. Risk-conscious correction of batch effects: maximising information extraction from high-throughput genomic datasets. *BMC bioinformatics*, 17:1–17, 2016.

[63] Vegard Nygaard, Einar Andreas Rødland, and Eivind Hovig. Methods that remove batch effects while retaining group differences may lead to exaggerated confidence in downstream analyses. *Biostatistics*, 17(1):29–39, 2016.

[64] Xinmin Li and Cun-Yu Wang. From bulk, single-cell to spatial rna sequencing. *International Journal of Oral Science*, 13(1):36, 2021.

[65] Vivien Marx. Method of the year: spatially resolved transcriptomics. *Nature methods*, 18(1):9–14, 2021.

[66] David S Fischer, Anna C Schaar, and Fabian J Theis. Modeling intercellular communication in tissues using spatial graphs of cells. *Nature Biotechnology*, 41(3):332–336, 2023.

[67] Mor Nitzan, Nikos Karaiskos, Nir Friedman, and Nikolaus Rajewsky. Gene expression cartography. *Nature*, 576(7785):132–137, 2019.

[68] Ruochen Jiang, Tianyi Sun, Dongyuan Song, and Jingyi Jessica Li. Statistics or biology: the zero-inflation controversy about scrna-seq data. *Genome biology*, 23(1):1–24, 2022.

[69] Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. Challenges in unsupervised clustering of single-cell rna-seq data. *Nature Reviews Genetics*, 20(5):273–282, 2019.

[70] Stephanie C Hicks, F William Townes, Mingxiang Teng, and Rafael A Irizarry. Missing data and technical variability in single-cell rna-sequencing experiments. *Biostatistics*, 19(4):562–578, 2018.

[71] Wei Vivian Li and Jingyi Jessica Li. An accurate and robust imputation method scimpute for single-cell rna-seq data. *Nature communications*, 9(1):997, 2018.

[72] David Van Dijk, Roshan Sharma, Juozas Nainys, Kristina Yim, Pooja Kathail, Ambrose J Carr, Cassandra Burdziak, Kevin R Moon, Christine L Chaffer, Diwakar Pattabiraman, et al. Recovering gene interactions from single-cell data using data diffusion. *Cell*, 174(3):716–729, 2018.

[73] Christopher S McGinnis, Lyndsay M Murrow, and Zev J Gartner. Doubletfinder: doublet detection in single-cell rna sequencing data using artificial nearest neighbors. *Cell systems*, 8(4):329–337, 2019.

[74] Samuel L Wolock, Romain Lopez, and Allon M Klein. Scrublet: computational identification of cell doublets in single-cell transcriptomic data. *Cell systems*, 8(4):281–291, 2019.

[75] Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Nature Precedings*, pages 1–1, 2010.

[76] Mark D Robinson, Davis J McCarthy, and Gordon K Smyth. edger: a bioconductor package for differential expression analysis of digital gene expression data. *bioinformatics*, 26(1):139–140, 2010.

[77] Mark D Robinson and Gordon K Smyth. Moderated statistical tests for assessing differences in tag abundance. *Bioinformatics*, 23(21):2881–2887, 2007.

[78] Purvesh Khatri, Marina Sirota, and Atul J Butte. Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS computational biology*, 8(2):e1002375, 2012.

[79] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[80] Suzi A Aleksander, James Balhoff, Seth Carbon, J Michael Cherry, Harold J Drabkin, Dustin Ebert, Marc Feuermann, Pascale Gaudet, Nomi L Harris, et al. The gene ontology knowledgebase in 2023. *Genetics*, 224(1):iyad031, 2023.

[81] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[82] Minoru Kanehisa, Michihiro Araki, Susumu Goto, Masahiro Hattori, Mika Hirakawa, Masumi Itoh, Toshiaki Katayama, Shuichi Kawashima, Shujiro Okuda, Toshiaki Tokimatsu, et al. Kegg for linking genomes to life and the environment. *Nucleic acids research*, 36(suppl_1):D480–D484, 2007.

[83] David E Gordon, Gwendolyn M Jang, Mehdi Bouhaddou, Jiewei Xu, Kirsten Obernier, Kris M White, Matthew J O'Meara, Veronica V Rezelj, Jeffrey Z Guo, Danielle L Swaney, et al. A sars-cov-2 protein interaction map reveals targets for drug repurposing. *Nature*, 583(7816):459–468, 2020.

[84] Qingguo Wang, Joshua Armenia, Chao Zhang, Alexander V Penson, Ed Reznik, Liguo Zhang, Thais Minet, Angelica Ochoa, Benjamin E Gross, Christine A Iacobuzio-Donahue, et al. Unifying cancer and normal RNA sequencing data from different sources. *Scientific data*, 5:180061, 2018.

[85] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[86] Andrey A Shabalin. Matrix eqtl: ultra fast eqtl analysis via large matrix operations. *Bioinformatics*, 28(10):1353–1358, 2012.

[87] Yoav Gilad, Scott A Rifkin, and Jonathan K Pritchard. Revealing the architecture of gene regulation: the promise of eqtl studies. *Trends in genetics*, 24(8):408–415, 2008.

[88] Clive J Hoggart, John C Whittaker, Maria De Iorio, and David J Balding. Simultaneous analysis of all snps in genome-wide and re-sequencing association studies. *PLoS genetics*, 4(7):e1000130, 2008.

[89] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

[90] Wolfgang Huber, Anja Von Heydebreck, Holger Sültmann, Annemarie Poustka, and Martin Vingron. Variance stabilization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, 18(suppl_1):S96–S104, 2002.

[91] Gordon K Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical applications in genetics and molecular biology*, 3(1), 2004.

[92] Bárbara Andrade Barbosa, Saskia D van Asten, Ji Won Oh, Arantza Farina-Sarasqueta, Joanne Verheij, Frederike Dijk, Hanneke WM van Laarhoven, Bauke Ylstra, Juan J Garcia Vallejo, Mark A van de Wiel, et al. Bayesian log-normal deconvolution for enhanced in silico microdissection of bulk gene expression data. *Nature communications*, 12(1):6106, 2021.

[93] Tinyi Chu, Zhong Wang, Dana Pe'er, and Charles G Danko. Cell type and gene expression deconvolution with bayesprism enables bayesian integrative analysis across bulk and single-cell rna sequencing in oncology. *Nature Cancer*, 3(4):505–517, 2022.

[94] Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

[95] Valentine Svensson, Adam Gayoso, Nir Yosef, and Lior Pachter. Interpretable factor models of single-cell rna-seq via variational autoencoders. *Bioinformatics*, 36(11):3418–3421, 2020.

[96] Vitalii Kleshchevnikov, Artem Shmatko, Emma Dann, Alexander Aivazidis, Hamish W King, Tong Li, Rasa Elmentaite, Artem Lomakin, Veronika Kedlian, Adam Gayoso, et al. Cell2location maps fine-grained cell types in spatial transcriptomics. *Nature biotechnology*, 40(5):661–671, 2022.

[97] John C Marioni, Christopher E Mason, Shrikant M Mane, Matthew Stephens, and Yoav Gilad. Rna-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome research*, 18(9):1509–1517, 2008.

[98] Peter V Kharchenko, Lev Silberstein, and David T Scadden. Bayesian approach to single-cell differential expression analysis. *Nature methods*, 11(7):740–742, 2014.

[99] Daniela M Witten. Classification and clustering of sequencing data using a poisson model. 2011.

[100] Yaqing Si, Peng Liu, Pinghua Li, and Thomas P Brutnell. Model-based clustering for rna-seq data. *Bioinformatics*, 30(2):197–205, 2014.

[101] Joseph M Hilbe. *Negative binomial regression*. Cambridge University Press, 2011.

[102] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):1–21, 2014.

[103] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E Kastriti, Peter Lönnerberg, Alessandro Furlan, et al. Rna velocity of single cells. *Nature*, 560(7719):494–498, 2018.

[104] Charity W Law, Yunshun Chen, Wei Shi, and Gordon K Smyth. voom: Precision weights unlock linear model analysis tools for rna-seq read counts. *Genome biology*, 15(2):1–17, 2014.

[105] Gökcen Eraslan, Lukas M Simon, Maria Mircea, Nikola S Mueller, and Fabian J Theis. Single-cell rna-seq denoising using a deep count autoencoder. *Nature communications*, 10(1):390, 2019.

[106] Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell rna-seq data. *Nature communications*, 9(1):284, 2018.

[107] Abhishek Sarkar and Matthew Stephens. Separating measurement and expression models clarifies confusion in single-cell rna sequencing analysis. *Nature genetics*, 53(6):770–777, 2021.

[108] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[109] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *CoRR*, abs/1906.02691, 2019.

[110] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[111] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[112] Ian J. Goodfellow. NIPS 2016 tutorial: Generative adversarial networks. *CoRR*, abs/1701.00160, 2017.

[113] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014.

[114] Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks, 2017.

[115] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan, 2017.

[116] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[117] Mark Schena, Dari Shalon, Ronald W. Davis, and Patrick O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270(5235):467–470, 1995.

[118] Rafael A Irizarry, Benjamin M Bolstad, Francois Collin, Leslie M Cope, Bridget Hobbs, and Terence P Speed. Summaries of affymetrix genechip probe level data. *Nucleic acids research*, 31(4):e15–e15, 2003.

[119] Ali Mortazavi, Brian A Williams, Kenneth McCue, Lorian Schaeffer, and Barbara Wold. Mapping and quantifying mammalian transcriptomes by RNA-seq. *Nat Meth*, 5(7):621–628, july 2008.

[120] Jing Yu, V. Anne Smith, Paul P. Wang, Alexander J. Hartemink, and Erich D. Jarvis. Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, 20(18):3594–3603, 2004.

[121] Adam A. Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7(1):S7, Mar 2006.

[122] Alexandre Irrthum, Louis Wehenkel, Pierre Geurts, et al. Inferring regulatory networks from expression data using tree-based methods. *PloS one*, 5(9):e12776, 2010.

[123] Tim Van den Bulcke, Koenraad Van Leemput, Bart Naudts, Piet van Remortel, Hongwu Ma, Alain Verschoren, Bart De Moor, and Kathleen Marchal. SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms. *BMC Bioinformatics*, 7(1):43, Jan 2006.

[124] Thomas Schaffter, Daniel Marbach, and Dario Floreano. GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, 27(16):2263–2270, 2011.

[125] Oliver Stegle, Leopold Parts, Matias Piipari, John Winn, and Richard Durbin. Using probabilistic estimation of expression residuals (peer) to obtain increased power and interpretability of gene expression analyses. *Nature protocols*, 7(3):500, 2012.

[126] Socorro Gama-Castro, Heladia Salgado, Alberto Santos-Zavaleta, Daniela Ledezma-Tejeida, Luis Muñiz-Rascado, Jair Santiago García-Sotelo, Kevin Alquicira-Hernández, Irma Martínez-Flores, Lucia Pannier, Jaime Abraham Castro-Mondragón, Alejandra Medina-Rivera, Hilda Solano-Lira, César Bonavides-Martínez, Ernesto Pérez-Rueda, Shirley Alquicira-Hernández, Liliana Porrón-Sotelo, Alejandra López-Fuentes, Anastasia Hernández-Koutoucheva, Víctor Del Moral-Chávez, Fabio Rinaldi, and Julio Collado-Vides. RegulonDB version

9.0: high-level integration of gene regulation, coexpression, motif clustering and beyond. *Nucleic Acids Research*, 44(D1):D133–D143, 2016.

[127] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv e-prints*, page arXiv:1701.07875, January 2017.

[128] Aladin Virmaux and Kevin Scaman. Lipschitz regularity of deep neural networks: analysis and efficient estimation. In *Advances in Neural Information Processing Systems*, pages 3835–3844, 2018.

[129] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[130] Jeremiah J. Faith, Michael E. Driscoll, Vincent A. Fusaro, Elissa J. Cosgrove, Boris Hayete, Frank S. Juhn, Stephen J. Schneider, and Timothy S. Gardner. Many microbe microarrays database: uniformly normalized affymetrix compendia with structured experimental metadata. *Nucleic Acids Research*, 36(suppl1):D866–D870, 2008.

[131] Heladia Salgado, Socorro Gama-Castro, Martín Peralta-Gil, Edgar Díaz-Peredo, Fabiola Sánchez-Solano, Alberto Santos-Zavaleta, Irma Martínez-Flores, Verónica Jiménez-Jacinto, César Bonavides-Martínez, Juan Segura-Salazar, Agustino Martínez-Antonio, and Julio Collado-Vides. RegulonDB (version 5.0): Escherichia coli k-12 transcriptional regulatory network, operon organization, and growth conditions. *Nucleic Acids Research*, 34(suppl_1):D394–D397, 2006.

[132] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, Cancer Genome Atlas Research Network, et al. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113, 2013.

[133] Matthew H Bailey, Collin Tokheim, Eduard Porta-Pardo, Sohini Sengupta, Denis Bertrand, Amila Weerasinghe, Antonio Colaprico, Michael C Wendl, Jaegil Kim, Brendan Reardon, et al. Comprehensive characterization of cancer driver genes and mutations. *Cell*, 173(2):371–385, 2018.

[134] Steffi Grote. GOfuncR: Gene ontology enrichment using FUNC. R package version 1.10.0. 2020.

[135] Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction, 2020.

[136] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu, and Jose M. Álvarez. Invertible conditional GANs for image editing. *CoRR*, abs/1611.06355, 2016.

[137] Grigory Antipov, Moez Baccouche, and Jean-Luc Dugelay. Face aging with conditional generative adversarial networks. *CoRR*, abs/1702.01983, 2017.

[138] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan, 2020.

[139] Sachith Gallolu Kankanamalage, Aroon S Karra, and Melanie H Cobb. Wnk pathways in cancer signaling networks. *Cell Communication and Signaling*, 16(1):1–6, 2018.

[140] Sónia Moniz and Peter Jordan. Emerging roles for wnk kinases in cancer. *Cellular and molecular life sciences*, 67(8):1265–1276, 2010.

[141] Jian Huang, Qing Deng, Qun Wang, Kun-Yu Li, Ji-Hong Dai, Niu Li, Zhi-Dong Zhu, Bo Zhou, Xiao-Yan Liu, Rui-Fang Liu, et al. Exome sequencing of hepatitis b virus–associated hepatocellular carcinoma. *Nature genetics*, 44(10):1117–1121, 2012.

[142] Naoshi Nishida and Masatoshi Kudo. Recent advancements in comprehensive genetic analyses for human hepatocellular carcinoma. *Oncology*, 84(Suppl. 1):93–97, 2013.

[143] Jiangyang Chi, Hongyan Zhang, Jia Hu, Yu Song, Jing Li, Lin Wang, and Zheng Wang. Agr3 promotes the stemness of colorectal cancer via modulating wnt/β-catenin signalling. *Cellular signalling*, 65:109419, 2020.

[144] Joanna Obacz, Martina Takacova, Veronika Brychtova, Petr Dobes, Silvia Pastorekova, Borivoj Vojtesek, and Roman Hrstka. The role of agr2 and agr3 in cancer: similar but not identical. *European journal of cell biology*, 94(3-4):139–147, 2015.

[145] Takashi Shimokawa, Satoshi Matsushima, Takuya Tsunoda, Hideaki Tahara, Yusuke Nakamura, and Yoichi Furukawa. Identification of tomm34, which shows elevated expression in the majority of human colon cancers, as a novel drug target. *International journal of oncology*, 29(2):381–386, 2006.

[146] Jose R Blesa, Jesus A Prieto-Ruiz, Beth A Abraham, Bridget L Harrison, Anita A Hegde, and Jose Hernandez-Yago. NRF-1 is the major transcription factor regulating the expression of the human TOMM34 gene. *Biochemistry and Cell Biology*, 86(1):46–56, 2008.

[147] Kiyotaka Okuno, Fumiaki Sugiura, Jin-ichi Hida, Tadao Tokoro, Eizaburo Ishimaru, Yasushi Sukegawa, and Kazuki Ueda. Phase i clinical trial of a novel peptide vaccine in combination with uft/lv for metastatic colorectal cancer. *ExperimEntal and therapEutic medicine*, 2(1):73–79, 2011.

[148] Regina Ebert, Sabine Zeck, Jutta Meissner-Weigl, Barbara Klotz, Tilman D Rachner, Peggy Benad, Ludger Klein-Hitpass, Maximilian Rudert, Lorenz C Hofbauer, and Franz Jakob. Krüppel-like factors klf2 and 6 and ki-67 are direct targets of zoledronic acid in mcf-7 cells. *Bone*, 50(3):723–732, 2012.

[149] CC Hu, YW Liang, JL Hu, LF Liu, JW Liang, and R Wang. Lncrna rusc1-as1 promotes the proliferation of breast cancer cells by epigenetic silence of klf2 and cdkn1a. *European review for medical and pharmacological sciences*, 23(15):6602–6611, 2019.

[150] JS Yu, S Koujak, S Nagase, CM Li, T Su, X Wang, M Keniry, L Memeo, A Rojtman, M Mansukhani, et al. Pcdh8, the human homolog of papc, is a candidate tumor suppressor of breast cancer. *Oncogene*, 27(34):4657–4665, 2008.

[151] Duyen H Pham, Chuan C Tan, Claire C Homan, Kristy L Kolc, Mark A Corbett, Dale McAninch, Archa H Fox, Paul Q Thomas, Raman Kumar, and Jozef Gecz. Protocadherin 19 (pcdh19) interacts with paraspeckle protein nono to co-regulate gene expression with estrogen receptor alpha (erα). *Human molecular genetics*, 26(11):2042–2052, 2017.

[152] Sumeet Dagar, Aparna Krishnadas, Israel Rubinstein, Michael J Blend, and Hayat Önyüksel. Vip grafted sterically stabilized liposomes for targeted imaging of breast cancer: in vivo studies. *Journal of Controlled Release*, 91(1-2):123–133, 2003.

[153] Hasan Zia, Toyoaki Hida, Sonia Jakowlew, Michael Birrer, Yehoshua Gozes, Jean C Reubi, Mati Fridkin, Illana Gozes, and Terry W Moody. Breast cancer growth is inhibited by vasoactive intestinal peptide (vip) hybrid, a synthetic vip receptor antagonist. *Cancer research*, 56(15):3486–3489, 1996.

[154] Terry W Moody, Joanna M Hill, and Robert T Jensen. Vip as a trophic factor in the cns and cancer cells. *Peptides*, 24(1):163–177, 2003.

[155] Graeme I Murray, Siva Patimalla, Keith N Stewart, Iain D Miller, and Steven D Heys. Profiling the expression of cytochrome p450 in breast cancer. *Histopathology*, 57(2):202–211, 2010.

[156] Isabelle Mercier, Mathew C Casimiro, Chenguang Wang, Anne L Rosenberg, Judy Quong, Alimatou Minkeu, Kathleen G Allen, Christiane Danilo, Federica Sotgia, Gloria Bonuccelli, et al. Human breast cancer-associated fibroblasts (cafs) show caveolin-1 down-regulation and rb tumor suppressor functional inactivation: implications for the response to hormonal therapy. *Cancer biology & therapy*, 7(8):1212–1225, 2008.

[157] Youn Soo Lee, Seon-Ah Ha, Hae Joo Kim, Seung Min Shin, Hyun Kee Kim, Sanghee Kim, Chang Suk Kang, Kyo Young Lee, Oak Kee Hong, Seung-Hwan Lee, et al. Minichromosome maintenance protein 3 is a candidate proliferation marker in papillary thyroid carcinoma. *Experimental and molecular pathology*, 88(1):138–142, 2010.

[158] Yusuf Ziya Igci, Suna Erkilic, Mehri Igci, and Ahmet Arslan. Mcm3 protein expression in follicular and classical variants of papillary thyroid carcinoma. *Pathology & Oncology Research*, 20(1):87–91, 2014.

[159] Mariya O Krisenko and Robert L Geahlen. Calling in syk: Syk's dual role as a tumor promoter and tumor suppressor in cancer. *Biochimica et Biophysica Acta (BBA)-Molecular Cell Research*, 1853(1):254–263, 2015.

[160] Ryan A Denu and Mark E Burkard. Analysis of the "centrosome-ome" identifies mcph1 deletion as a cause of centrosome amplification in human cancer. *Scientific Reports*, 10(1):1–17, 2020.

[161] Yinglian Pan, Li Ping Jia, Yuzhu Liu, Yixu Han, and Qingchun Deng. Alteration of tumor associated neutrophils by pik3ca expression in endometrial carcinoma from tcga data. *Journal of ovarian research*, 12(1):81, 2019.

[162] Michael J Demeure, Meraj Aziz, Richard Rosenberg, Steven D Gurley, Kimberly J Bussey, and John D Carpten. Whole-genome sequencing of an aggressive braf wild-type papillary thyroid cancer identified eml4–alk translocation as a therapeutic target. *World journal of surgery*, 38(6):1296–1305, 2014.

[163] Hyun Joo Kim, Young Ha Kim, Dong Soo Lee, June-Key Chung, and Soonhag Kim. In vivo imaging of functional targeting of mir-221 in papillary thyroid carcinoma. *Journal of Nuclear Medicine*, 49(10):1686–1693, 2008.

[164] Yuan Lu, Jilong Li, Jianlin Cheng, and Dennis B Lubahn. Messenger rna profile analysis deciphers new esrrb responsive genes in prostate cancer cells. *BMC molecular biology*, 16(1):21, 2015.

[165] Rachel N Winter, Andrew Kramer, Andrew Borkowski, and Natasha Kyprianou. Loss of caspase-1 and caspase-3 protein expression in human prostate cancer. *Cancer research*, 61(3):1227–1232, 2001.

[166] Sudheer K Mantena, Som D Sharma, and Santosh K Katiyar. Berberine, a natural product, induces g1-phase cell cycle arrest and caspase-3-dependent apoptosis in human prostate carcinoma cells. *Molecular cancer therapeutics*, 5(2):296–308, 2006.

[167] Jing-Ding Wang, Shiro Takahara, Norio Nonomura, Naotsugu Ichimaru, Kiyohide Toki, Haruhito Azuma, Kiyomi Matsumiya, Akihiko Okuyama, and Seiichi Suzuki. Early induction of apoptosis in androgen-independent prostate cancer cell line by fty720 requires caspase-3 activation. *The Prostate*, 40(1):50–55, 1999.

[168] Yongliang Yang, S James Adelstein, and Amin I Kassis. Putative molecular signatures for the imaging of prostate cancer. *Expert review of molecular diagnostics*, 10(1):65–74, 2010.

[169] Ana Cheong, Xiang Zhang, Yuk-Yin Cheung, Wan-yee Tang, Jing Chen, Shu-Hua Ye, Mario Medvedovic, Yuet-Kin Leung, Gail S Prins, and Shuk-Mei Ho. DNA methylome changes by estradiol benzoate and bisphenol a links early-life environmental exposures to prostate cancer risk. *Epigenetics*, 11(9):674–689, 2016.

[170] Mary E Gagou, Anil Ganesh, Geraldine Phear, Darren Robinson, Eva Petermann, Angela Cox, and Mark Meuth. Human pif1 helicase supports dna replication and cell growth under oncogenic-stress. *Oncotarget*, 5(22):11381, 2014.

[171] William Cookson, Liming Liang, Gonçalo Abecasis, Miriam Moffatt, and Mark Lathrop. Mapping complex disease traits with global gene expression. *Nature Reviews Genetics*, 10(3):184–194, 2009.

[172] Valur Emilsson, Gudmar Thorleifsson, Bin Zhang, Amy S Leonardson, Florian Zink, Jun Zhu, Sonia Carlson, Agnar Helgason, G Bragi Walters, Steinunn Gunnarsdottir, et al. Genetics of gene expression and its effect on disease. *Nature*, 452(7186):423–428, 2008.

[173] Eric R Gamazon, Ayellet V Segrè, Martijn van de Bunt, Xiaoquan Wen, Hualin S Xi, Farhad Hormozdiari, Halit Ongen, Anuar Konkashbaev, Eske M Derks, François Aguet, et al. Using an atlas of gene regulation across 44 human tissues to inform complex disease-and trait-associated variation. *Nature genetics*, 50(7):956–967, 2018.

[174] Marina Sirota, Joel T Dudley, Jeewon Kim, Annie P Chiang, Alex A Morgan, Alejandro Sweet-Cordero, Julien Sage, and Atul J Butte. Discovery and preclinical validation of drug indications using compendia of public gene expression data. *Science translational medicine*, 3(96):96ra77–96ra77, 2011.

[175] William E Evans and Mary V Relling. Moving towards individualized medicine with pharmacogenomics. *Nature*, 429(6990):464–468, 2004.

[176] Mary-Claire King and Allan C Wilson. Evolution at two levels in humans and chimpanzees. *Science*, 188(4184):107–116, 1975.

[177] Laura L Colbran, Eric R Gamazon, Dan Zhou, Patrick Evans, Nancy J Cox, and John A Capra. Inferred divergent gene regulation in archaic hominins reveals potential phenotypic differences. *Nature ecology & evolution*, 3(11):1598–1606, 2019.

[178] Lucie A Low, Christine Mummery, Brian R Berridge, Christopher P Austin, and Danilo A Tagle. Organs-on-chips: into the next decade. *Nature Reviews Drug Discovery*, pages 1–17, 2020.

[179] Anvita Gupta and James Zou. Feedback gan for dna optimizes protein functions. *Nature Machine Intelligence*, 1(2):105–111, 2019.

[180] Michael E Wall, Andreas Rechtsteiner, and Luis M Rocha. Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis*, pages 91–109. Springer, 2003.

[181] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

[182] Øystein Sørensen, Kristoffer Herland Hellton, Arnoldo Frigessi, and Magne Thoresen. Covariate selection in high-dimensional generalized linear models with measurement error. *Journal of Computational and Graphical Statistics*, 27(4):739–749, 2018.

[183] GTEx Consortium et al. Genetic effects on gene expression across human tissues. *Nature*, 550(7675):204–213, 2017.

[184] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. GAIN: Missing data imputation using generative adversarial nets. *arXiv preprint arXiv:1806.02920*, 2018.

[185] Mohamed Marouf, Pierre Machart, Vikas Bansal, Christoph Kilian, Daniel S Magruder, Christian F Krebs, and Stefan Bonn. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nature communications*, 11(1):1–12, 2020.

[186] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. 2014.

[187] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[188] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 448–456. JMLR.org, 2015.

[189] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*, volume 793. John Wiley & Sons, 2019.

[190] Kayoung Kim, Min-Ji Kim, Su Yeong Kim, Steve Park, Chan Beum Park, et al. Clinically accurate diagnosis of alzheimer's disease via multiplexed sensing of core biomarkers in human plasma. *Nature communications*, 11(1):1–9, 2020.

[191] Lorenzo Beretta and Alessandro Santaniello. Nearest neighbor imputation algorithms: a critical evaluation. *BMC medical informatics and decision making*, 16(3):74, 2016.

[192] S van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, pages 1–68, 2010.

[193] Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

[194] Kara L Carter, Ellen Cahir-McFarland, and Elliott Kieff. Epstein-barr virus-induced changes in b-lymphocyte gene expression. *Journal of virology*, 76(20):10427–10436, 2002.

[195] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3(29):861, 2018.

[196] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[197] Cancer Genome Atlas Research Network. Genomic and epigenomic landscapes of adult de novo acute myeloid leukemia. *New England Journal of Medicine*, 368(22):2059–2074, 2013.

[198] Cancer Genome Atlas Network et al. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61, 2012.

[199] Cancer Genome Atlas Research Network et al. Comprehensive molecular profiling of lung adenocarcinoma. *Nature*, 511(7511):543–550, 2014.

[200] Stephen B Baylin and Peter A Jones. A decade of exploring the cancer epigenome—biological and translational implications. *Nature Reviews Cancer*, 11(10):726–734, 2011.

[201] John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, Cancer Genome Atlas Research Network, et al. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113, 2013.

[202] Haiyan Huang, Chun-Chi Liu, and Xianghong Jasmine Zhou. Bayesian approach to transforming public gene expression repositories into disease diagnosis databases. *Proceedings of the National Academy of Sciences*, 107(15):6823–6828, 2010.

[203] Xiaochen Sun, Santiago Vilar, and Nicholas P Tatonetti. High-throughput methods for combinatorial drug discovery. *Science translational medicine*, 5(205):205rv1–205rv1, 2013.

[204] Ahmed Hosny and Hugo JWL Aerts. Artificial intelligence for global health. *Science*, 366(6468):955–956, 2019.

[205] Danton S Char, Nigam H Shah, and David Magnus. Implementing machine learning in health care—addressing ethical challenges. *The New England journal of medicine*, 378(11):981, 2018.

[206] Genevieve L Wojcik, Mariaelisa Graff, Katherine K Nishimura, Ran Tao, Jeffrey Haessler, Christopher R Gignoux, Heather M Highland, Yesha M Patel, Elena P Sorokin, Christy L Avery, et al. Genetic analyses of diverse populations improves discovery for complex traits. *Nature*, 570(7762):514–518, 2019.

[207] Dan Zhou, Yi Jiang, Xue Zhong, Nancy J Cox, Chunyu Liu, and Eric R Gamazon. A unified framework for joint-tissue transcriptome-wide association and mendelian randomization analysis. *Nature Genetics*, 2020.

[208] Xiaonan Yang, Ling Kui, Min Tang, Dawei Li, Kunhua Wei, Wei Chen, Jianhua Miao, and Yang Dong. High-throughput transcriptome profiling in drug and biomarker discovery. *Frontiers in genetics*, 11:19, 2020.

[209] Chenling Xu, Romain Lopez, Edouard Mehlman, Jeffrey Regier, Michael I Jordan, and Nir Yosef. Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models. *Molecular systems biology*, 17(1):e9620, 2021.

[210] Dave SB Hoon, Peter Bostick, Christine Kuo, Tetsuro Okamoto, He-Jing Wang, Robert Elashoff, and Donald L Morton. Molecular markers in blood as surrogate prognostic indicators of melanoma recurrence. *Cancer research*, 60(8):2253–2257, 2000.

[211] Chaochao Cai, Peter Langfelder, Tova F Fuller, Michael C Oldham, Rui Luo, Leonard H van den Berg, Roel A Ophoff, and Steve Horvath. Is human blood a good surrogate for brain tissue in transcriptional studies? *BMC genomics*, 11(1):1–15, 2010.

[212] Geoffrey Istas, Ken Declerck, Maria Pudenz, Veronica Lendinez-Tortajada, Montserrat Leon-Latre, Karen Heyninck, Guy Haegeman, Jose A Casasnovas, Maria Tellez-Plaza, Clarissa Gerhauser, et al. Identification of differentially methylated brca1 and crisp2 dna regions as blood surrogate markers for cardiovascular disease. *Scientific reports*, 7(1):1–14, 2017.

[213] Jiebiao Wang, Eric R Gamazon, Brandon L Pierce, Barbara E Stranger, Hae Kyung Im, Robert D Gibbons, Nancy J Cox, Dan L Nicolae, and Lin S Chen. Imputing gene expression in uncollected tissues within and beyond gtex. *The American Journal of Human Genetics*, 98(4):697–708, 2016.

[214] Jae Hoon Sul, Buhm Han, Chun Ye, Ted Choi, and Eleazar Eskin. Effectively identifying eqtls from multiple tissues by combining mixed model and meta-analytic approaches. *PLoS genetics*, 9(6):e1003491, 2013.

[215] Gokcen Eraslan, Eugene Drokhlyansky, Shankara Anand, Ayshwarya Subramanian, Evgenij Fiskin, Michal Slyper, Jiali Wang, Nicholas Van Wittenberghe, John M. Rouhana, Julia Waldman, Orr Ashenberg, Danielle Dionne, Thet Su Win, Michael S. Cuoco, Olena Kuksenko, Philip A. Branton, Jamie L. Marshall, Anna Greka, Gad Getz, Ayellet V. Segrè, François Aguet, Orit Rozenblatt-Rosen, Kristin G. Ardlie, and Aviv Regev. Single-nucleus cross-tissue molecular reference maps to decipher disease gene function. *bioRxiv*, 2021.

[216] Jean-Philippe Brunet, Pablo Tamayo, Todd R Golub, and Jill P Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the national academy of sciences*, 101(12):4164–4169, 2004.

[217] Soumya Raychaudhuri, Joshua M Stuart, and Russ B Altman. Principal components analysis to summarize microarray experiments: application to sporulation time series. In *Biocomputing 2000*, pages 455–466. World Scientific, 1999.

[218] Jiaxuan You, Xiaobai Ma, Daisy Ding, Mykel Kochenderfer, and Jure Leskovec. Handling missing data with graph representation learning. *NeurIPS*, 2020.

[219] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

[220] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. *arXiv preprint arXiv:2006.05205*, 2020.

[221] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks?, 2022.

[222] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

[223] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[224] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[225] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28 Issue 1, 2014.

[226] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International conference on machine learning*, pages 2071–2080. PMLR, 2016.

[227] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32 Issue 1, 2018.

[228] Komal Teru, Etienne Denis, and Will Hamilton. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning*, pages 9448–9457. PMLR, 2020.

[229] Dobrik Georgiev, Marc Brockschmidt, and Miltiadis Allamanis. HEAT: hyperedge attention networks. *CoRR*, abs/2201.12113, 2022.

[230] Mohammadamin Tavakoli, Alexander Shmakov, Francesco Ceccarelli, and Pierre Baldi. Rxn hypergraph: a hypergraph attention model for chemical reaction representation. *arXiv preprint arXiv:2201.01196*, 2022.

[231] F Alexander Wolf, Philipp Angerer, and Fabian J Theis. Scanpy: large-scale single-cell gene expression data analysis. *Genome biology*, 19(1):1–5, 2018.

[232] Tim Stuart, Andrew Butler, Paul Hoffman, Christoph Hafemeister, Efthymia Papalexi, William M Mauck III, Yuhan Hao, Marlon Stoeckius, Peter Smibert, and Rahul Satija. Comprehensive integration of single-cell data. *Cell*, 177(7):1888–1902, 2019.

[233] Valentine Svensson, Adam Gayoso, Nir Yosef, and Lior Pachter. Interpretable factor models of single-cell RNA-seq via variational autoencoders. *Bioinformatics*, 36(11):3418–3421, 03 2020.

[234] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[235] Till Roenneberg and Martha Merrow. The circadian clock and human health. *Current biology*, 26(10):R432–R443, 2016.

[236] Jean-Michel Davière and Patrick Achard. Organ communication: Cytokinins on the move. *Nature plants*, 3(8):1–2, 2017.

[237] Sue C Bodine, Heddwen L Brooks, Nigel W Bunnett, Hilary A Coller, Mark R Frey, Bina Joe, Thomas R Kleyman, Merry L Lindsey, Andre Marette, Rory E Morty, et al. An american physiological society cross-journal call for papers on "inter-organ communication in homeostasis and disease", 2021.

[238] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[239] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[240] Sandip Ray, Markus Britschgi, Charles Herbert, Yoshiko Takeda-Uchimura, Adam Boxer, Kaj Blennow, Leah Friedman, Douglas Galasko, Marek Jutel, Anna Karydas, Jeffrey Kaye, Jerzy Leszek, Bruce Miller, Lennart Minthon, Joseph Quinn, Gil Rabinovici, William Robinson, Marwan Sabbagh, Yuen So, and Tony Wyss-Coray. Classification and prediction of clinical alzheimer's diagnosis based on plasma signaling proteins. *Nature Medicine*, 13:1359–1362, 10 2007.

[241] Kasper Lage, Niclas Tue Hansen, E Olof Karlberg, Aron C Eklund, Francisco S Roque, Patricia K Donahoe, Zoltan Szallasi, Thomas Skøt Jensen, and Søren Brunak. A large-scale analysis of tissue-specific pathology and gene expression of human disease genes and complexes. *Proceedings of the National Academy of Sciences*, 105(52):20870–20875, 2008.

[242] Hélène-Marie Lanoiselée, Gaël Nicolas, David Wallon, Anne Rovelet-Lecrux, Morgane Lacour, Stéphane Rousseau, Anne-Claire Richard, Florence Pasquier, Adeline Rollin-Sillaire, Olivier Martinaud, et al. App, psen1, and psen2 mutations in early-onset alzheimer disease: A genetic screening study of familial and sporadic cases. *PLoS medicine*, 14(3):e1002270, 2017.

[243] Lynn M Bekris, Chang-En Yu, Thomas D Bird, and Debby W Tsuang. Genetics of alzheimer disease. *Journal of geriatric psychiatry and neurology*, 23(4):213–227, 2010.

[244] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[245] Urmo Võsa, Annique Claringbould, Harm-Jan Westra, Marc Jan Bonder, Patrick Deelen, Biao Zeng, Holger Kirsten, Ashis Saha, Roman Kreuzhuber, Seyhan Yazar, et al. Large-scale cis-and trans-eqtl analyses identify thousands of genetic loci and polygenic scores that regulate blood gene expression. *Nature genetics*, 53(9):1300–1310, 2021.

[246] Daifeng Wang, Shuang Liu, Jonathan Warrell, Hyejung Won, Xu Shi, Fabio CP Navarro, Declan Clarke, Mengting Gu, Prashant Emani, Yucheng T Yang, et al. Comprehensive functional genomic resource and integrative model for the human brain. *Science*, 362(6420):eaat8464, 2018.

[247] Ryan Tewhey, Dylan Kotliar, Daniel S Park, Brandon Liu, Sarah Winnicki, Steven K Reilly, Kristian G Andersen, Tarjei S Mikkelsen, Eric S Lander, Stephen F Schaffner, et al. Direct identification of hundreds of expression-modulating variants using a multiplexed reporter assay. *Cell*, 165(6):1519–1529, 2016.

[248] Alexandra C Nica and Emmanouil T Dermitzakis. Expression quantitative trait loci: present and future. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 368(1620):20120362, 2013.

[249] Matthew V Rockman and Leonid Kruglyak. Genetics of global gene expression. *Nature Reviews Genetics*, 7(11):862–872, 2006.

[250] John D Storey and Robert Tibshirani. Statistical significance for genomewide studies. *Proceedings of the National Academy of Sciences*, 100(16):9440–9445, 2003.

[251] Samuel Davis, Thomas H Aldrich, David M Valenzuela, Vivien Wong, Mark E Furth, Stephen P Squinto, and George D Yancopoulos. The receptor for ciliary neurotrophic factor. *Science*, 253(5015):59–63, 1991.

[252] Baoji Xu and Xiangyang Xie. Neurotrophic factor control of satiety and body weight. *Nature Reviews Neuroscience*, 17(5):282–292, 2016.

[253] Maxim V Kuleshov, Matthew R Jones, Andrew D Rouillard, Nicolas F Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L Jenkins, Kathleen M Jagodnik, Alexander Lachmann, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.

[254] Clair R Martin, Vadim Osadchiy, Amir Kalani, and Emeran A Mayer. The brain-gut-microbiome axis. *Cellular and molecular gastroenterology and hepatology*, 6(2):133–148, 2018.

[255] Sumei Liu. Neurotrophic factors in enteric physiology and pathophysiology. *Neurogastroenterology & Motility*, 30(10):e13446, 2018.

[256] Louis J Sparvero, Denise Asafu-Adjei, Rui Kang, Daolin Tang, Neilay Amin, Jaehyun Im, Ronnye Rutledge, Brenda Lin, Andrew A Amoscato, Herbert J Zeh, et al. Rage (receptor for advanced glycation endproducts), rage ligands, and their

role in cancer and inflammation. *Journal of translational medicine*, 7(1):1–21, 2009.

[257] Yifan Zhao, Huiyu Cai, Zuobai Zhang, Jian Tang, and Yue Li. Learning interpretable cellular and gene signature embeddings from single-cell transcriptomic data. *Nature communications*, 12(1):1–15, 2021.

[258] Heonjong Han, Hongseok Shim, Donghyun Shin, Jung Eun Shim, Yunhee Ko, Junha Shin, Hanhae Kim, Ara Cho, Eiru Kim, Tak Lee, et al. Trrust: a reference database of human transcriptional regulatory interactions. *Scientific reports*, 5(1):1–11, 2015.

[259] Larysa Pevny, M Celeste Simon, Elizabeth Robertson, William H Klein, Shih-Feng Tsai, Vivette D'Agati, Stuart H Orkin, and Frank Costantini. Erythroid differentiation in chimaeric mice blocked by a targeted mutation in the gene for transcription factor gata-1. *Nature*, 349(6306):257–260, 1991.

[260] Andrew D Sharrocks. The ets-domain transcription factor family. *Nature reviews Molecular cell biology*, 2(11):827–837, 2001.

[261] Angela Wedel and HW Lömsziegler-Heitbrock. The c/ebp family of transcription factors. *Immunobiology*, 193(2-4):171–185, 1995.

[262] Claus Nerlov. The c/ebp family of transcription factors: a paradigm for interaction between gene expression and proliferation control. *Trends in cell biology*, 17(7):318–324, 2007.

[263] Chilakamarti V Ramana, Moitreyee Chatterjee-Kishore, Hannah Nguyen, and George R Stark. Complex roles of stat1 in regulating gene expression. *Oncogene*, 19(21):2619–2627, 2000.

[264] Claus Nerlov, Erich Querfurth, Holger Kulessa, and Thomas Graf. Gata-1 interacts with the myeloid pu. 1 transcription factor and represses pu. 1-dependent transcription. *Blood, The Journal of the American Society of Hematology*, 95(8):2543–2551, 2000.

[265] Kosuke Zenke, Masashi Muroi, and Ken-ichi Tanamoto. Irf1 supports dna binding of stat1 by promoting its phosphorylation. *Immunology and cell biology*, 96(10):1095–1103, 2018.

[266] Alexander Lachmann, Zhuorui Xie, and Avi Ma'ayan. blitzgsea: efficient computation of gene set enrichment analysis through gamma distribution approximation. *Bioinformatics*, 38(8):2356–2357, 2022.

[267] Dylan Kotliar, Adrian Veres, M Aurel Nagy, Shervin Tabrizi, Eran Hodis, Douglas A Melton, and Pardis C Sabeti. Identifying gene expression programs of cell-type identity and cellular activity with single-cell rna-seq. *Elife*, 8, 2019.

[268] Hugo Gonzalez, Catharina Hagerling, and Zena Werb. Roles of the immune system in cancer: from tumor initiation to metastatic progression. *Genes & development*, 32(19-20):1267–1284, 2018.

[269] Marc Elosua-Bayes, Paula Nieto, Elisabetta Mereu, Ivo Gut, and Holger Heyn. SPOTlight: seeded NMF regression to deconvolute spatial transcriptomics spots with single-cell transcriptomes. *Nucleic Acids Research*, 49(9):e50–e50, 02 2021.

[270] Dylan M. Cable, Evan Murray, Luli S. Zou, Aleksandrina Goeva, Evan Z. Macosko, Fei Chen, and Rafael A. Irizarry. Robust decomposition of cell type mixtures in spatial transcriptomics. *Nature Biotechnology*, 40(4):517–526, Apr 2022.

[271] Rui Dong and Guo-Cheng Yuan. Spatialdwls: accurate deconvolution of spatial transcriptomic data. *Genome Biology*, 22(1):145, May 2021.

[272] Felix Wu, Tianyi Zhang, Amauri H. Souza Jr., Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. *CoRR*, abs/1902.07153, 2019.

[273] Paul Scherer, Helena Andrés-Terré, Pietro Liò, and Mateja Jamnik. Decoupling feature propagation from the design of graph auto-encoders. *CoRR*, abs/1910.08589, 2019.

[274] Jeffrey R. Moffitt, Dhananjay Bambah-Mukku, Stephen W. Eichhorn, Eric Vaughn, Karthik Shekhar, Julio D. Perez, Nimrod D. Rubinstein, Junjie Hao, Aviv Regev, Catherine Dulac, and Xiaowei Zhuang. Molecular, spatial, and functional single-cell profiling of the hypothalamic preoptic region. *Science*, 362(6416):eaau5324, 2018.

[275] Amanda Janesick, Robert Shelansky, Andrew D Gottscho, Florian Wagner, Morgane Rouault, Ghezal Beliakoff, Michelli Faria de Oliveira, Andrew Kohlway, Jawad Abousoud, Carolyn A Morrison, et al. High resolution mapping of the breast cancer tumor microenvironment using integrated single cell, spatial and in situ analysis of ffpe tissue. *bioRxiv*, pages 2022–10, 2022.

[276] Monika Piwecka, Nikolaus Rajewsky, and Agnieszka Rybak-Wolf. Single-cell and spatial transcriptomics: deciphering brain complexity in health and disease. *Nature Reviews Neurology*, pages 1–17, 2023.

[277] Niyaz Yoosuf, José Fernández Navarro, Fredrik Salmén, Patrik L Ståhl, and Carsten O Daub. Identification and transfer of spatial transcriptomics signatures for cancer diagnosis. *Breast Cancer Research*, 22:1–10, 2020.

[278] Linlin Zhang, Dongsheng Chen, Dongli Song, Xiaoxia Liu, Yanan Zhang, Xun Xu, and Xiangdong Wang. Clinical and translational values of spatial transcriptomics. *Signal Transduction and Targeted Therapy*, 7(1):111, 2022.

[279] Michael Wainberg, Nasa Sinnott-Armstrong, Nicholas Mancuso, Alvaro N Barbeira, David A Knowles, David Golan, Raili Ermel, Arno Ruusalepp, Thomas Quertermous, Ke Hao, et al. Opportunities and challenges for transcriptome-wide association studies. *Nature genetics*, 51(4):592–599, 2019.

[280] Mohammad Lotfollahi, Mohsen Naghipourfar, Fabian J Theis, and F Alexander Wolf. Conditional out-of-distribution generation for unpaired data using transfer vae. *Bioinformatics*, 36(Supplement_2):i610–i617, 2020.

[281] Vishwali Mhasawade, Yuan Zhao, and Rumi Chunara. Machine learning and algorithmic fairness in public and population health. *Nature Machine Intelligence*, 3(8):659–666, 2021.

[282] Céline Bellenguez, Fahri Küçükali, Iris E Jansen, Luca Kleineidam, Sonia Moreno-Grau, Najaf Amin, Adam C Naj, Rafael Campos-Martin, Benjamin Grenier-Boley, Victor Andrade, et al. New insights into the genetic etiology of alzheimer's disease and related dementias. *Nature genetics*, 54(4):412–436, 2022.

[283] Catherine Tcheandjieu, Xiang Zhu, Austin T Hilliard, Shoa L Clarke, Valerio Napolioni, Shining Ma, Kyung Min Lee, Huaying Fang, Fei Chen, Yingchang Lu, et al. Large-scale genome-wide association study of coronary artery disease in genetically diverse populations. *Nature medicine*, 28(8):1679–1692, 2022.

[284] Amit V Khera, Mark Chaffin, Krishna G Aragam, Mary E Haas, Carolina Roselli, Seung Hoan Choi, Pradeep Natarajan, Eric S Lander, Steven A Lubitz, Patrick T Ellinor, et al. Genome-wide polygenic scores for common diseases identify individuals with risk equivalent to monogenic mutations. *Nature genetics*, 50(9):1219–1224, 2018.

[285] Emil Uffelmann, Qin Qin Huang, Nchangwi Syntia Munung, Jantina De Vries, Yukinori Okada, Alicia R Martin, Hilary C Martin, Tuuli Lappalainen, and Danielle Posthuma. Genome-wide association studies. *Nature Reviews Methods Primers*, 1(1):59, 2021.

[286] Judea Pearl. *Causality*. Cambridge university press, 2009.

[287] Martin Jinek, Krzysztof Chylinski, Ines Fonfara, Michael Hauer, Jennifer A Doudna, and Emmanuelle Charpentier. A programmable dual-rna–guided dna endonuclease in adaptive bacterial immunity. *science*, 337(6096):816–821, 2012.

[288] Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell rna profiling of pooled genetic screens. *cell*, 167(7):1853–1866, 2016.

[289] Yusuf Roohani, Kexin Huang, and Jure Leskovec. Gears: Predicting transcriptional outcomes of novel multi-gene perturbations. *BioRxiv*, pages 2022–07, 2022.

[290] Rui Chen and Michael Snyder. Promise of personalized omics to precision medicine. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 5(1):73–82, 2013.

[291] Katherine A Overmyer, Evgenia Shishkova, Ian J Miller, Joseph Balnis, Matthew N Bernstein, Trenton M Peters-Clarke, Jesse G Meyer, Qiuwen Quan, Laura K Muehlbauer, Edna A Trujillo, et al. Large-scale multi-omic analysis of covid-19 severity. *Cell systems*, 12(1):23–40, 2021.

[292] Aravind Subramanian, Rajiv Narayan, Steven M Corsello, David D Peck, Ted E Natoli, Xiaodong Lu, Joshua Gould, John F Davis, Andrew A Tubelli, Jacob K Asiedu, et al. A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell*, 171(6):1437–1452, 2017.

[293] Xiaoping Han, Renying Wang, Yincong Zhou, Lijiang Fei, Huiyu Sun, Shujing Lai, Assieh Saadatpour, Ziming Zhou, Haide Chen, Fang Ye, et al. Mapping the mouse cell atlas by microwell-seq. *Cell*, 172(5):1091–1107, 2018.

[294] Christina V Theodoris, Ling Xiao, Anant Chopra, Mark D Chaffin, Zeina R Al Sayed, Matthew C Hill, Helene Mantineo, Elizabeth M Brydon, Zexian Zeng, X Shirley Liu, et al. Transfer learning enables predictions in network biology. *Nature*, pages 1–9, 2023.

[295] Zhi-Jie Cao and Ge Gao. Multi-omics single-cell data integration and regulatory inference with graph-linked embedding. *Nature Biotechnology*, 40(10):1458–1466, 2022.

[296] Maria Brbić, Marinka Zitnik, Sheng Wang, Angela O Pisco, Russ B Altman, Spyros Darmanis, and Jure Leskovec. Mars: discovering novel cell types across heterogeneous single-cell experiments. *Nature methods*, 17(12):1200–1206, 2020.

[297] Johan Ludwig William Valdemar Jensen et al. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta mathematica*, 30:175–193, 1906.

[298] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[299] Lukas Biewald. Experiment tracking with weights and biases, 2020. Software available from wandb.com.

[300] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.

[301] Pierre-Alexandre Mattei and Jes Frellsen. Miwae: Deep generative modelling and imputation of incomplete data sets. In *International conference on machine learning*, pages 4413–4423. PMLR, 2019.

[302] Stef Van Buuren. *Flexible imputation of missing data*. CRC press, 2018.

[303] James J Heckman. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. In *Annals of economic and social measurement, volume 5, number 4*, pages 475–492. NBER, 1976.

[304] Robert J Glynn, Nan M Laird, and Donald B Rubin. Selection modeling versus mixture modeling with nonignorable nonresponse. In *Drawing inferences from self-selected samples*, pages 115–142. Springer, 1986.

[305] Joseph L Schafer and John W Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002.

[306] Linda M Collins, Joseph L Schafer, and Chi-Ming Kam. A comparison of inclusive and restrictive strategies in modern missing data procedures. *Psychological methods*, 6(4):330, 2001.

[307] Rita Ferreira, Kinuko Ohneda, Masayuki Yamamoto, and Sjaak Philipsen. Gata1 function, a paradigm for transcription factors in hematopoiesis. *Molecular and cellular biology*, 25(4):1215–1227, 2005.

[308] Leonidas C Platanias. Mechanisms of type-i-and type-ii-interferon-mediated signalling. *Nature Reviews Immunology*, 5(5):375–386, 2005.

[309] Data Coordinating Center Burton Robert 67 Jensen Mark A 53 Kahn Ari 53 Pihl Todd 53 Pot David 53 Wan Yunhu 53 and Tissue Source Site Levine Douglas A 68. The cancer genome atlas pan-cancer analysis project. *Nature genetics*, 45(10):1113–1120, 2013.

[310] Qingguo Wang, Joshua Armenia, Chao Zhang, Alexander V Penson, Ed Reznik, Liguo Zhang, Thais Minet, Angelica Ochoa, Benjamin E Gross, Christine A Iacobuzio-Donahue, et al. Unifying cancer and normal rna sequencing data from different sources. *Scientific data*, 5(1):1–8, 2018.

[311] Dvir Aran, Roman Camarda, Justin Odegaard, Hyojung Paik, Boris Oskotsky, Gregor Krings, Andrei Goga, Marina Sirota, and Atul J Butte. Comprehensive analysis of normal adjacent to tumor transcriptomes. *Nature communications*, 8(1):1077, 2017.

[312] Ignacio Vázquez-García, Florian Uhlitz, Nicholas Ceglia, Jamie LP Lim, Michelle Wu, Neeman Mohibullah, Juliana Niyazov, Arvin Eric B Ruiz, Kevin M Boehm, Viktoria Bojilova, et al. Ovarian cancer mutational processes drive site-specific immune evasion. *Nature*, pages 1–9, 2022.

Supplementary Information A

Generative models

A.1 ELBO derivation

To derive the ELBO, we first expand the log likelihood via the marginalisation rule and introduce an auxiliary, variational distribution q_ϕ:

$$\begin{aligned}
\log p_\theta(\mathbf{x}) &= \log \sum_{\mathbf{z}} p_\theta(\mathbf{x}, \mathbf{z}) \\
&= \log \sum_{\mathbf{z}} \frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})} p_\theta(\mathbf{x}, \mathbf{z}) \\
&= \log \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \right]
\end{aligned}$$

Since the logarithm is a concave function, we can use the Jensen's inequality $\log(\mathbb{E}[\cdot]) \geq \mathbb{E}[\log(\cdot)]$ [297] to move the logarithm inward, obtaining the evidence lower bound $\mathcal{L}_{ELBO}$:

$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \right] = \mathcal{L}_{ELBO}$$

Finally, we rewrite the ELBO in its standard form as follows:

$$\begin{aligned}
\mathcal{L}_{ELBO} &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})} \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})} \right] \\
&= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} [\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{KL}\Big(q_\phi(\mathbf{z}|\mathbf{x}) || p_\theta(\mathbf{z}) \Big)
\end{aligned}$$

A.2 Generative adversarial imputation nets

An interesting adaptation of the GAN framework that I studied for the purpose of imputing missing data are Generative Adversarial Imputation Nets (GAINs; [184]). In this framework, the generator imputes the missing components of the input based on the observed values, while the discriminator takes imputed samples as input and attempts to distinguish whether each component has been observed or produced by the generator. This is in contrast to the original GAN discriminator, which receives information from two input streams (generator and data distribution) and attempts to distinguish the true input source.

The generator aims at implicitly estimating the distribution $\mathbb{P}_{\mathbf{x}|\tilde{\mathbf{x}},\mathbf{m}}$, representing the probability of $\mathbf{x}$ given a binary mask of missing components $\mathbf{m}$ and a noisy view of $\mathbf{x}$, which we denote as $\tilde{\mathbf{x}}$, wherein the missing components have been masked out (e.g. $\tilde{\mathbf{x}} = \mathbf{x} \odot \mathbf{m}$). Therefore, its role is not only to impute missing components, but also to reconstruct the observed inputs. Let n be the number of input variables. Formally, the generator is a function $G_\theta : \mathbb{R}^n \times \mathbb{R}^n \times \{0,1\}^n \to \mathbb{R}^n$ that produces a vector of imputed values $\bar{\mathbf{x}}$ as follows:

$$\bar{\mathbf{x}} = G_\theta(\mathbf{x} \odot \mathbf{m}, \mathbf{z} \odot (\mathbf{1} - \mathbf{m}), \mathbf{m}),$$

where the noise vector $\mathbf{z}$ is masked as $\mathbf{z} \odot (\mathbf{1} - \mathbf{m})$ to encourage a bijective association between noise components and input variables. Before passing the output $\bar{\mathbf{x}}$ to the discriminator, [184] replace the prediction for the non-missing components by the original, observed values:

$$\hat{\mathbf{x}} = \mathbf{m} \odot \tilde{\mathbf{x}} + (\mathbf{1} - \mathbf{m}) \odot \bar{\mathbf{x}}$$

The discriminator takes the imputed samples $\hat{\mathbf{x}}$ and attempts to distinguish whether the expression value of each gene has been observed or produced by the generator. Formally, the discriminator is a function $D_\omega : \mathbb{R}^n \times \mathbb{R}^n \to [0,1]^n$ that outputs the probabilities $\hat{\mathbf{y}}$ of each value being observed as opposed to being imputed by the generator:

$$\hat{\mathbf{y}} = D_\omega(\hat{\mathbf{x}}, \mathbf{h})$$

Here, the vector $\mathbf{h} \in \mathbb{R}^n$ corresponds to the *hint* mechanism described in [184], which provides theoretical guarantees on the uniqueness of the global minimum for the estimation of $\mathbb{P}_{\mathbf{x}|\tilde{\mathbf{x}},\mathbf{m}}$. Concretely, the role of the hint vector $\mathbf{h}$ is to *leak* some

information about the mask $\mathbf{m}$ to the discriminator. The hint $\mathbf{h}$ is defined as follows:

$$\mathbf{h} = \mathbf{b} \odot \mathbf{m} + \frac{1}{2}(\mathbf{1} - \mathbf{b}) \qquad \mathbf{b} \sim \mathbb{B}(1, p), \tag{A.1}$$

where $\mathbf{b} \in \{0, 1\}^n$ is a binary vector sampled from a Bernoulli distribution $\mathbb{B}(1, p)$ with probability p, which controls the amount of information from the mask $\mathbf{m}$ revealed to the discriminator. The model is optimised via the following minimax game:

$$\min_{\theta} \max_{\omega} \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{m}\sim\mathbb{P}_r,\mathbf{b}\sim\mathbb{B}(1,p),\mathbf{z}\sim\mathbb{P}_{\mathbf{z}}} \left[\mathbf{m}^\top \log \hat{\mathbf{y}} + (1 - \mathbf{m})^\top \log \left(\mathbf{1} - \hat{\mathbf{y}}\right)\right]$$

Supplementary Information B

In-silico generation of tissue-specific gene expression

B.1 Example dendrogrammatic distances

The coefficient $\gamma(C(\mathbf{D}^X), C(\mathbf{D}^Z))$ does not necessarily correlate well with $\gamma(\mathbf{D}^X, \mathbf{D}^Z)$. Consider for example the distance matrices:

$$\mathbf{D}^X = \begin{bmatrix} 0 & 2 & 10 \\ 2 & 0 & 3 \\ 10 & 3 & 0 \end{bmatrix} \quad \mathbf{D}^Z = \begin{bmatrix} 0 & 3 & 10 \\ 3 & 0 & 2 \\ 10 & 2 & 0 \end{bmatrix} \tag{B.1}$$

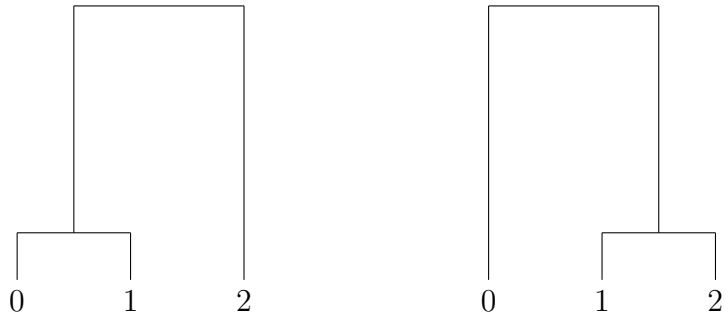
The dendrogrammatic distance matrices $C(\mathbf{D}^X)$ and $C(\mathbf{D}^Z)$ resulting from agglomerative hierarchical clustering with complete linkage are:

$$C(\mathbf{D}^X) = \begin{bmatrix} 0 & 2 & 10 \\ 2 & 0 & 10 \\ 10 & 10 & 0 \end{bmatrix} \quad C(\mathbf{D}^Z) = \begin{bmatrix} 0 & 10 & 10 \\ 10 & 0 & 2 \\ 10 & 2 & 0 \end{bmatrix} \tag{B.2}$$

And the coefficients $\gamma(\mathbf{D}^X, \mathbf{D}^Z) = 0.97$ and $\gamma(C(\mathbf{D}^X), C(\mathbf{D}^Z)) = -0.5$ are substantially different. Figure B.1 illustrates these dendrograms.

B.2 SynTReN validation scores

We selected the noise hyperparameters that optimise the S_{dist} score on the train set.



(a) Dendrogram for $C(\mathbf{D}^X)$

(b) Dendrogram for $C(\mathbf{D}^Z)$

Fig. B.1 Dendrograms resulting from agglomerative hierachical clustering with complete linkage for the distance matrices $\mathbf{D}^X$ and $\mathbf{D}^Z$ defined in equation B.1. Note that $\gamma(\mathbf{D}^X, \mathbf{D}^Z) = 0.97$, but $\gamma(C(\mathbf{D}^X), C(\mathbf{D}^Z)) = -0.5$ because the dendrograms' structures are substantially different.

B.3 GeneNetWeaver validation scores

We produced multifactorial experiments using the default settings for the DREAM4 network inference challenge (`http://gnw.sourceforge.net/dreamchallenge.html`). We selected the noise term that optimise the S_{dist} score on the train set.

Table B.1 Validation scores for different configurations of the SynTReN noise hyperparameters

Biological noise	Experimental noise	S_{dist}	S_{dend}	$S_{\text{TF-TG}}$	$S_{\text{TG-TG}}$
0.0	0.0	0.0370	0.0221	0.1705	0.2438
0.0	0.1	0.0364	0.0239	0.1794	0.2433
0.0	0.2	0.0351	0.0281	0.1910	0.2430
0.0	0.5	0.0300	0.0307	0.2131	0.2312
0.0	0.8	0.0267	0.0273	0.2158	0.2101
0.1	0.0	0.0384	0.0330	0.1842	0.2531
0.1	0.1	0.0379	0.0312	0.1888	0.2522
0.1	0.2	0.0363	0.0263	0.1967	0.2507
0.1	0.5	0.0311	0.0286	0.2129	0.2373
0.1	0.8	0.0276	0.0238	0.2166	0.2156
0.2	0.0	0.0399	0.0315	0.1963	0.2666
0.2	0.1	0.0394	0.0363	0.1984	0.2653
0.2	0.2	0.0381	0.0313	0.2027	0.2626
0.2	0.5	0.0332	0.0323	0.2126	0.2472
0.2	0.8	0.0295	0.0287	0.2155	0.2247
0.5	0.0	0.0448	0.0492	0.2035	0.2842
0.5	0.1	0.0447	0.0446	0.2043	0.2831
0.5	0.2	0.0441	0.0422	0.2060	0.2803
0.5	0.5	0.0411	0.0440	0.2118	0.2654
0.8	0.0	0.0498	0.0536	0.2001	0.2784
0.8	0.1	0.0498	0.0475	0.2000	0.2779
0.8	0.2	0.0495	0.0504	0.2007	0.2764
0.8	0.5	0.0476	0.0495	0.2049	0.2669
0.8	0.8	0.0449	0.0417	0.2116	0.2521

Table B.2 Validation scores for different configurations of the GNW noise hyperparameter

Noise term	S_{dist}	S_{dend}	$S_{\text{TF-TG}}$	$S_{\text{TG-TG}}$
0	0.0569	0.0344	0.1591	0.1876
0.05	0.0605	0.0329	0.1596	0.1929
0.1	0.0645	0.0236	0.1828	0.2068
0.2	0.0508	0.0309	0.2112	0.2036
0.5	0.0454	0.0211	0.1953	0.1851
0.8	0.0298	0.0087	0.2147	0.1394

B.4 Supplementary figures

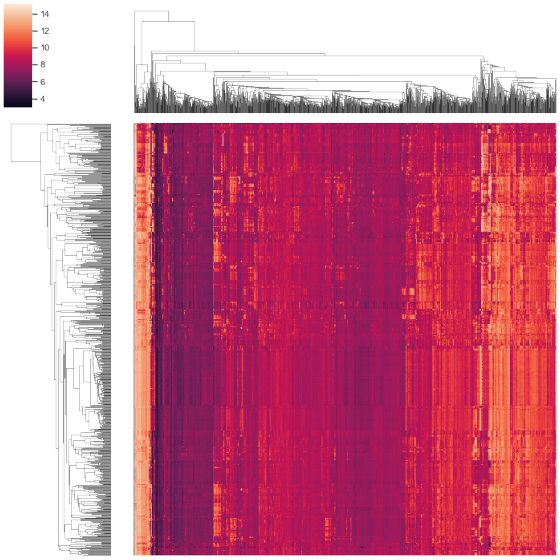


Fig. B.2 Clustering *E. coli* gene expression data for the *E. coli* M^{3D} dataset (CRP hierarchy).

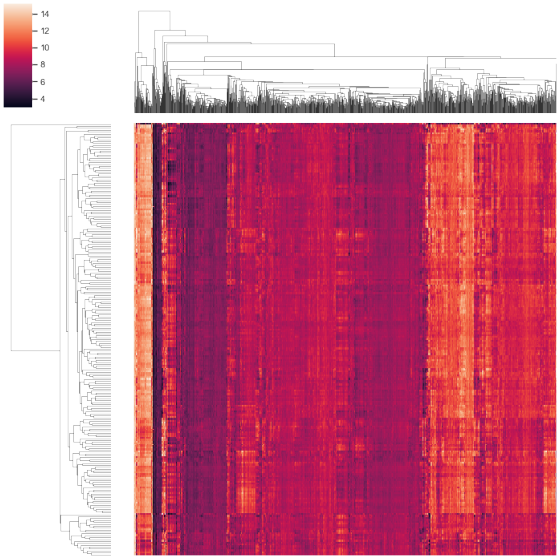


Fig. B.3 Clustering *E. coli* gene expression data for the dataset generated with the GAN on the CRP hierarchy.

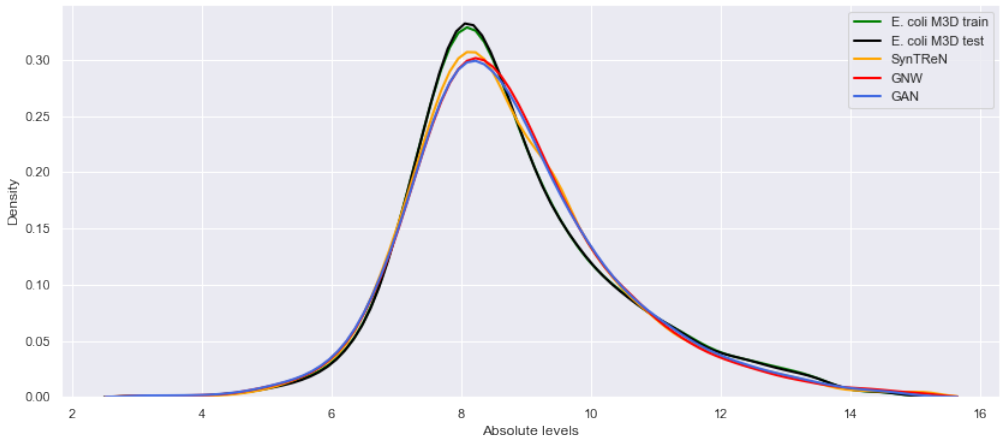


Fig. B.4 Distribution of gene intensities.

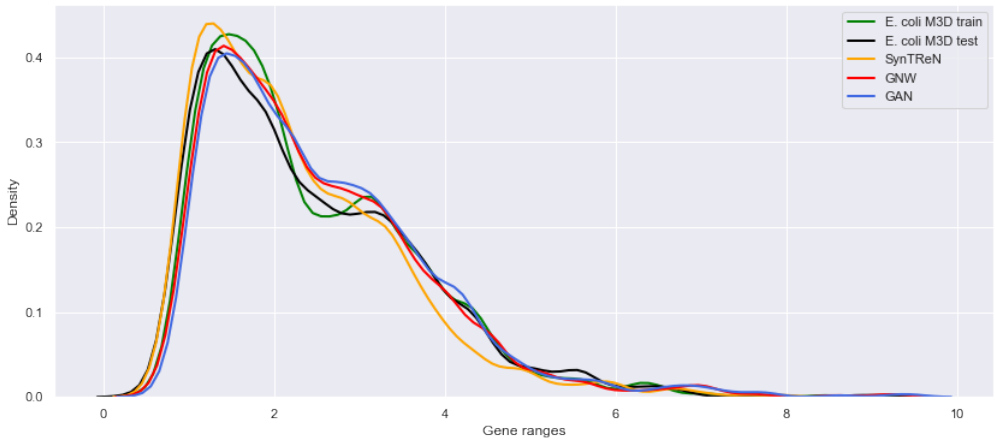


Fig. B.5 Distribution of gene expression ranges.

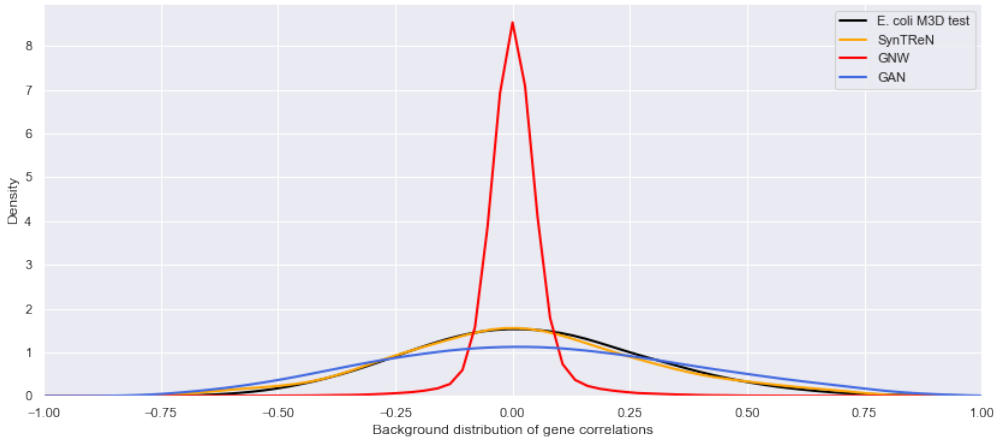


Fig. B.6 Background distribution of the Pearson's correlation coefficients between all pair of genes for SynTReN, GNW, and GAN.

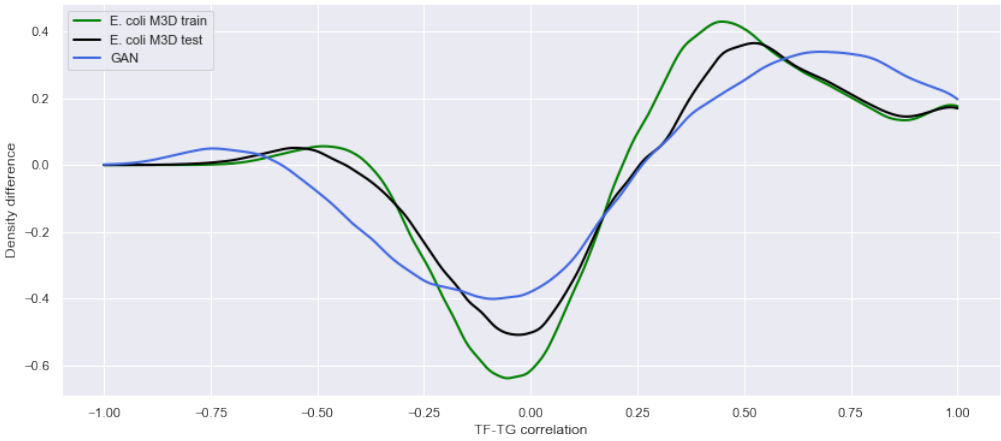


Fig. B.7 Histogram of TF-TG interactions. It shows to what extent TF-TG pairs are enriched (> 0) or depleted (< 0) with respect to the background distribution.

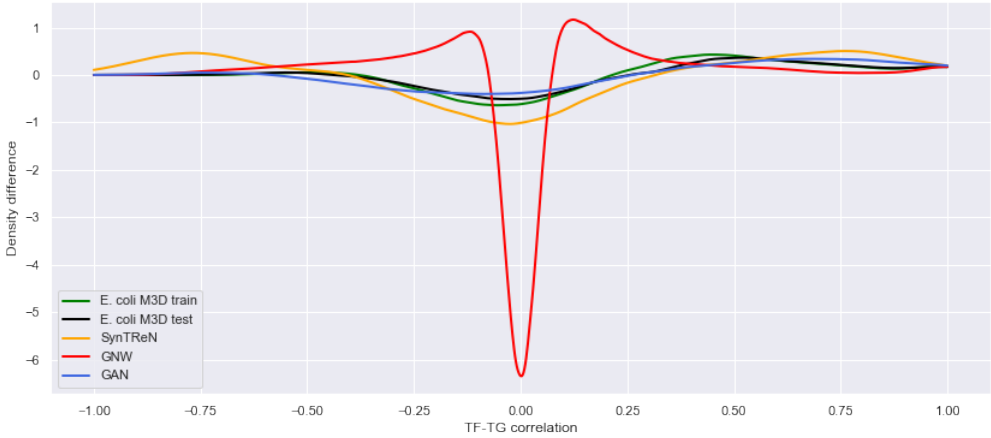


Fig. B.8 Histogram of TF-TG interactions (including SynTReN and GNW). It shows to what extent TF-TG pairs are enriched (> 0) or depleted (< 0) with respect to the background distribution.

Fig. B.9 Histogram of TG-TG interactions. It shows to what extent TG-TG pairs are enriched (> 0) or depleted (< 0) with respect to the background distribution.

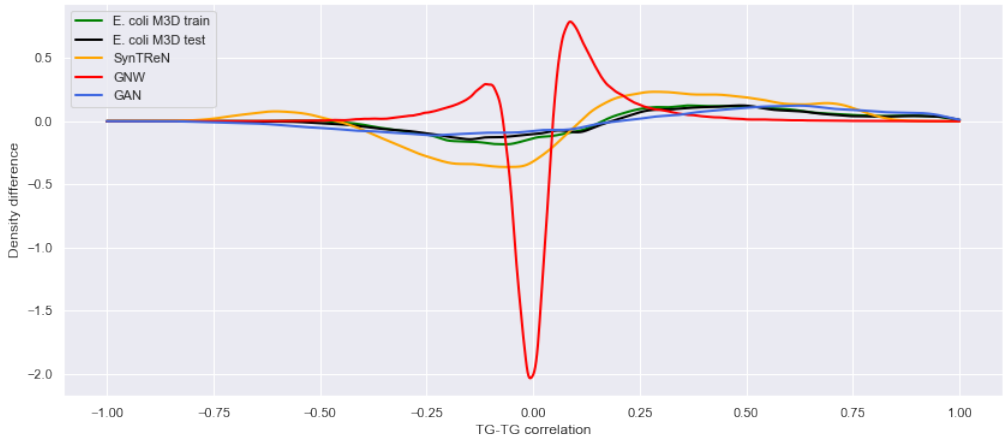


Fig. B.10 Histogram of TG-TG interactions (including SynTReN and GNW). It shows to what extent TG-TG pairs are enriched (> 0) or depleted (< 0) with respect to the background distribution.

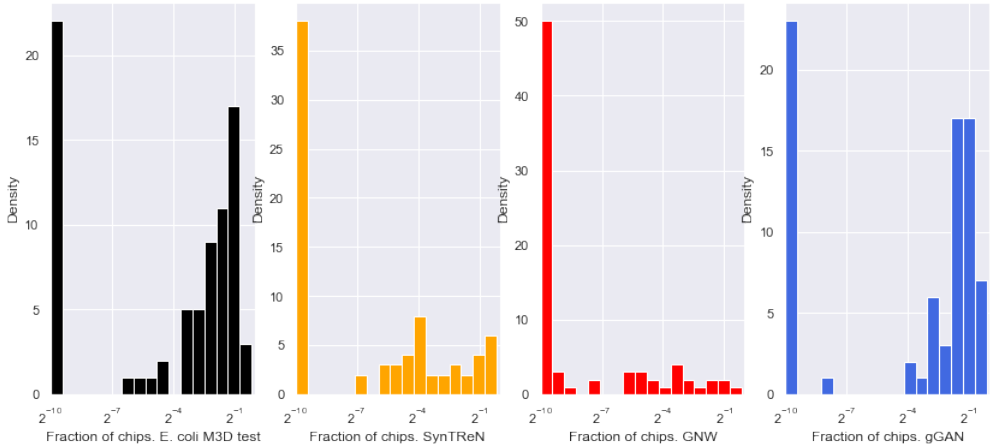


Fig. B.11 Histograms of the TF activity (including SynTReN and GNW). They are formed by computing the fraction of samples in which TF targets exhibit rank differences with respect to other non TF targets, according to a two-sided Mann-Whitney rank test. These tests are corrected with the Benjamini-Hochberg's procedure in order to account for multiple testing and reduce the false discovery rate.

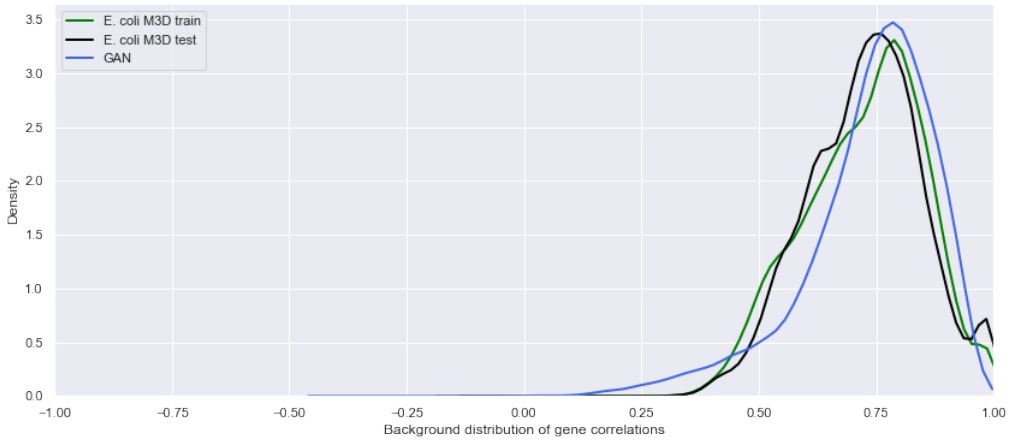


Fig. B.12 Background distribution of sample correlations. This plot allows us to check whether "mode collapse" occurs. Mode collapse is a well-known problem of GANs where the generator outputs samples from a few, limited set of modes that are realistic to the critic. In the extreme case, the generator would always output the same sample and therefore all the sample pairwise correlations would be close to one.

B.5 Table of enriched Gene Ontology terms per cluster

The following table shows the enriched Gene Ontology terms for each pair of matching clusters in Figure 3. For each cluster, we show the enriched terms with a family-wise error rate (FWER) smaller than 0.05. Gene Ontology terms highlighted in bold have a $FWER < 0.05$ in both matching clusters. We used the R package GOfuncR [134].

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
1	**GO:1901564**	organonitrogen compound metabolic process	0.000	**GO:0009056**	catabolic process	0.001
	GO:0044237	cellular metabolic process	0.000	**GO:0044248**	cellular catabolic process	0.002
	GO:0036211	protein modification process	0.000	**GO:1901575**	organic substance catabolic process	0.003
	GO:0006464	cellular protein modification process	0.000	GO:0048522	positive regulation of cellular process	0.003
	GO:0033036	macromolecule localization	0.000	**GO:1901564**	organonitrogen compound metabolic process	0.004
	GO:0071705	nitrogen compound transport	0.000	GO:0048518	positive regulation of biological process	0.006
	GO:0009058	biosynthetic process	0.000	**GO:0044237**	cellular metabolic process	0.006
	GO:0071702	organic substance transport	0.000	**GO:0044260**	cellular macromolecule metabolic process	0.007
	GO:0051234	establishment of localization	0.000	GO:0006796	phosphate-containing compound metabolic process	0.007
	GO:1901575	organic substance catabolic process	0.000	GO:0006793	phosphorus metabolic process	0.009
	GO:0051179	localization	0.000	**GO:0044238**	primary metabolic process	0.011
	GO:0044260	cellular macromolecule metabolic process	0.000	**GO:0016043**	cellular component organization	0.013
	GO:0043412	macromolecule modification	0.000	**GO:0033036**	macromolecule localization	0.014

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:1901576	organic substance biosynthetic process	0.001	**GO:0006807**	nitrogen compound metabolic process	0.014
	GO:0044249	cellular biosynthetic process	0.001	GO:0080090	regulation of primary metabolic process	0.016
	GO:0006810	transport	0.001	GO:0051171	regulation of nitrogen compound metabolic process	0.017
	GO:0009056	catabolic process	0.001	**GO:0009058**	biosynthetic process	0.019
	GO:0033554	cellular response to stress	0.001	**GO:0071840**	cellular component organization or biogenesis	0.021
	GO:0023051	regulation of signaling	0.001	**GO:1901576**	organic substance biosynthetic process	0.032
	GO:0044238	primary metabolic process	0.001	GO:0031323	regulation of cellular metabolic process	0.042
	GO:0071840	cellular component organization or biogenesis	0.001			
	GO:0006807	nitrogen compound metabolic process	0.001			
	GO:0015031	protein transport	0.001			
	GO:0042886	amide transport	0.001			
	GO:0019538	protein metabolic process	0.001			
	GO:0008104	protein localization	0.001			
	GO:0045184	establishment of protein localization	0.001			

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0016043	cellular component organization	0.003			
	GO:0015833	peptide transport	0.003			
	GO:0010646	regulation of cell communication	0.003			
	GO:1901565	organonitrogen compound catabolic process	0.006			
	GO:0048856	anatomical structure development	0.012			
	GO:0023056	positive regulation of signaling	0.012			
	GO:0044267	cellular protein metabolic process	0.026			
	GO:0032502	developmental process	0.027			
	GO:0006950	response to stress	0.027			
	GO:0009987	cellular process	0.037			
	GO:0009966	regulation of signal transduction	0.045			
	GO:0010647	positive regulation of cell communication	0.045			
	GO:0044248	cellular catabolic process	0.046			
	GO:0050790	regulation of catalytic activity	0.049			

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
2	GO:0023056	positive regulation of signaling	0.000	**GO:0051179**	localization	0.003
	GO:0010647	positive regulation of cell communication	0.000	**GO:0048518**	positive regulation of biological process	0.025
	GO:0048518	positive regulation of biological process	0.000	**GO:0065009**	regulation of molecular function	0.031
	GO:1902533	positive regulation of intracellular signal transduction	0.000			
	GO:0051179	localization	0.000			
	GO:0065009	regulation of molecular function	0.000			
	GO:0046903	secretion	0.000			
	GO:0010033	response to organic substance	0.000			
	GO:0009967	positive regulation of signal transduction	0.000			
	GO:0032940	secretion by cell	0.000			
	GO:0048583	regulation of response to stimulus	0.003			
	GO:0048584	positive regulation of response to stimulus	0.003			
	GO:0032879	regulation of localization	0.004			

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0140352	export from cell	0.004			
	GO:0045055	regulated exocytosis	0.005			
	GO:0023051	regulation of signaling	0.006			
	GO:0010646	regulation of cell communication	0.008			
	GO:0009605	response to external stimulus	0.014			
	GO:0006810	transport	0.014			
	GO:0065008	regulation of biological quality	0.015			
	GO:0051649	establishment of localization in cell	0.015			
	GO:0051234	establishment of localization	0.024			
	GO:0048522	positive regulation of cellular process	0.028			
	GO:0071310	cellular response to organic substance	0.030			
	GO:0070887	cellular response to chemical stimulus	0.033			
	GO:0006887	exocytosis	0.035			
	GO:0044248	cellular catabolic process	0.038			
3	**GO:0044260**	cellular macromolecule metabolic process	0.000	**GO:0044260**	cellular macromolecule metabolic process	0.000

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0051171	regulation of nitrogen compound metabolic process	0.000	**GO:0051641**	cellular localization	0.001
	GO:0065008	regulation of biological quality	0.000	**GO:0071840**	cellular component organization or biogenesis	0.006
	GO:0051173	positive regulation of nitrogen compound metabolic process	0.001	**GO:0009893**	positive regulation of metabolic process	0.010
	GO:0009893	positive regulation of metabolic process	0.001	**GO:0072657**	protein localization to membrane	0.013
	GO:0031325	positive regulation of cellular metabolic process	0.001	GO:0071705	nitrogen compound transport	0.013
	GO:0080090	regulation of primary metabolic process	0.002	**GO:0051171**	regulation of nitrogen compound metabolic process	0.014
	GO:0031323	regulation of cellular metabolic process	0.002	**GO:0080090**	regulation of primary metabolic process	0.014
	GO:0009058	biosynthetic process	0.002	**GO:0031325**	positive regulation of cellular metabolic process	0.015
	GO:0009059	macromolecule biosynthetic process	0.002	**GO:0031323**	regulation of cellular metabolic process	0.027
	GO:1901576	organic substance biosynthetic process	0.005			
	GO:0044249	cellular biosynthetic process	0.005			

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0048522	positive regulation of cellular process	0.005			
	GO:0010604	positive regulation of macromolecule metabolic process	0.007			
	GO:0034645	cellular macromolecule biosynthetic process	0.007			
	GO:0048518	positive regulation of biological process	0.007			
	GO:0071840	cellular component organization or biogenesis	0.008			
	GO:0006807	nitrogen compound metabolic process	0.008			
	GO:0072657	protein localization to membrane	0.010			
	GO:0051641	cellular localization	0.010			
	GO:0044271	cellular nitrogen compound biosynthetic process	0.035			
	GO:0044238	primary metabolic process	0.041			
	GO:0051252	regulation of RNA metabolic process	0.045			
4	GO:0051649	establishment of localization in cell	0.004			

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0051641	cellular localization	0.004			
	GO:0015833	peptide transport	0.016			
	GO:0015031	protein transport	0.016			
	GO:0042886	amide transport	0.017			
	GO:0140352	export from cell	0.017			
	GO:0071705	nitrogen compound transport	0.022			
	GO:0045184	establishment of protein localization	0.039			
	GO:0071702	organic substance transport	0.049			
5	**GO:0010033**	response to organic substance	0.000	GO:0043412	macromolecule modification	0.000
	GO:0044260	cellular macromolecule metabolic process	0.000	**GO:0010033**	response to organic substance	0.001
	GO:1901576	organic substance biosynthetic process	0.002	**GO:0044260**	cellular macromolecule metabolic process	0.002
	GO:0010604	positive regulation of macromolecule metabolic process	0.003	GO:0016043	cellular component organization	0.004
	GO:0044249	cellular biosynthetic process	0.003	GO:0023051	regulation of signaling	0.004
	GO:0080090	regulation of primary metabolic process	0.004	GO:0071840	cellular component organization or biogenesis	0.004

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0031323	regulation of cellular metabolic process	0.004	**GO:0010604**	positive regulation of macromolecule metabolic process	0.006
	GO:0019219	regulation of nucleobase-containing compound metabolic process	0.006	GO:0051641	cellular localization	0.006
	GO:0009058	biosynthetic process	0.012	GO:0010646	regulation of cell communication	0.007
	GO:0051173	positive regulation of nitrogen compound metabolic process	0.019	GO:0006464	cellular protein modification process	0.009
	GO:0009059	macromolecule biosynthetic process	0.019	GO:0036211	protein modification process	0.009
	GO:0051171	regulation of nitrogen compound metabolic process	0.024	GO:0019538	protein metabolic process	0.012
	GO:0034645	cellular macromolecule biosynthetic process	0.026	GO:0033554	cellular response to stress	0.017
	GO:0010243	response to organonitrogen compound	0.046	GO:0002224	toll-like receptor signaling pathway	0.018
				GO:0051128	regulation of cellular component organization	0.027
				GO:0070887	cellular response to chemical stimulus	0.033

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
				GO:0051649	establishment of localization in cell	0.040
6	GO:0050794	regulation of cellular process	0.015			
7	**GO:0032787**	monocarboxylic acid metabolic process	0.013	GO:0044260	cellular macromolecule metabolic process	0.002
				GO:0032787	monocarboxylic acid metabolic process	0.004
				GO:1901564	organonitrogen compound metabolic process	0.004
				GO:1901576	organic substance biosynthetic process	0.005
				GO:0044237	cellular metabolic process	0.005
				GO:0009058	biosynthetic process	0.008
				GO:0044249	cellular biosynthetic process	0.015
				GO:0048523	negative regulation of cellular process	0.020
				GO:0001676	long-chain fatty acid metabolic process	0.037
8				GO:0051173	positive regulation of nitrogen compound metabolic process	0.000
				GO:0031325	positive regulation of cellular metabolic process	0.001

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
				GO:0009893	positive regulation of metabolic process	0.003
				GO:0009058	biosynthetic process	0.003
				GO:0051247	positive regulation of protein metabolic process	0.003
				GO:0010604	positive regulation of macromolecule metabolic process	0.003
				GO:0044260	cellular macromolecule metabolic process	0.003
				GO:0032270	positive regulation of cellular protein metabolic process	0.003
				GO:1901564	organonitrogen compound metabolic process	0.007
				GO:1901576	organic substance biosynthetic process	0.008
				GO:0044249	cellular biosynthetic process	0.008
				GO:0044093	positive regulation of molecular function	0.012
				GO:0032984	protein-containing complex disassembly	0.013
				GO:0043624	cellular protein complex disassembly	0.016

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
				GO:0051179	localization	0.019
				GO:0071840	cellular component organization or biogenesis	0.024
				GO:0051234	establishment of localization	0.024
				GO:0006810	transport	0.038
				GO:0048522	positive regulation of cellular process	0.040
				GO:0010942	positive regulation of cell death	0.042
				GO:0008219	cell death	0.047
9	**GO:0007275**	multicellular organism development	0.000	**GO:0044260**	cellular macromolecule metabolic process	0.001
	GO:0071840	cellular component organization or biogenesis	0.003	**GO:0007275**	multicellular organism development	0.001
	GO:0007399	nervous system development	0.003	**GO:0071840**	cellular component organization or biogenesis	0.001
	GO:0048856	anatomical structure development	0.003	**GO:0048856**	anatomical structure development	0.001
	GO:1901564	organonitrogen compound metabolic process	0.011	GO:0006996	organelle organization	0.007
	GO:0032502	developmental process	0.022	**GO:1901564**	organonitrogen compound metabolic process	0.008

Cluster	Real cluster			Matching synthetic cluster		
	GO term	GO name	FWER	GO term	GO name	FWER
	GO:0044260	cellular macromolecule metabolic process	0.026	**GO:0016043**	cellular component organization	0.013
	GO:0016043	cellular component organization	0.037	GO:0045944	positive regulation of transcription by RNA polymerase II	0.013
	GO:0048731	system development	0.042	**GO:0032502**	developmental process	0.017
				GO:0045893	positive regulation of transcription, DNA-templated	0.028
				GO:0051254	positive regulation of RNA metabolic process	0.041
				GO:0007155	cell adhesion	0.044
				GO:0045935	positive regulation of nucleobase-containing compound metabolic process	0.049
10	GO:1901564	organonitrogen compound metabolic process	0.022			
	GO:0044238	primary metabolic process	0.041			

Supplementary Information C

Intra-tissue imputation of gene expression

C.1 Observations about GAIN's adversarial loss

We implemented the adversarial loss of Generative Adversarial Imputation Networks (GAIN) as described in the GAIN paper [184]. Our implementation can be found at: `https://github.com/rvinas/GAIN-GTEx`. Our results show that the effects of the adversarial loss on the R^2 imputation scores are small or negligible. We have investigated this issue in great detail and our observations are the following:

- One hypothesis is that the dimensionality of the gene expression data might be too high for GAIN. This was also discussed in a Github issue (`https://github.com/jsyoon0823/GAIN/issues/9`). For the Alzheimer's disease pathway case study (273 genes) and the in-place scenario, including the adversarial term seems to yield a small improvement in the R^2 scores. Nonetheless, the scores are fairly similar for the other scenarios.

- The weights for the adversarial and mean squared error (MSE) terms might not be properly adjusted. However, when we set the MSE weight to 0, the model failed to converge and the R^2 results were very poor. Without the MSE loss, the training was unstable in all our experiments. Additionally, as described in a Github issue (`https://github.com/jsyoon0823/GAIN/issues/8`), decreasing the weight of the MSE term (e.g., from 1 to 0.1) leads to slower convergence.

- The adversarial loss might be incompatible with certain features of the model or hyperparameter configurations. However, different hyperparameters (including

batch normalisation, dropout, and number of hidden units per layer) led to a similar performance with and without adversarial loss.

- The discriminator and generator might need to be well balanced, that is, the discriminator might require more gradient updates to learn useful representations of the data. This idea was also discussed in a Github issue (`https://github.com/jsyoon0823/GAIN/issues/17`), where it is also argued that the model is very sensitive to different hyperparameter configurations. However, after several experiments (e.g., we trained the discriminator more often than the generator), we did not observe significant improvements relative to using the MSE loss exclusively.

For the purpose of reproducibility, as the gains of the adversarial loss appear to be small or negligible given our observations, we recommend training GAIN-GTEx without the adversarial term.

C.2 Scalability analysis for MissForest

Figures C.1 and C.2 show the runtime of *a single* iteration of the MissForest algorithm [193] as we vary the number of samples and genes. We fixed the number of trees to 3 and the maximum depth per tree to 3.

Figure C.3 shows the runtime of MissForest for a subset as we vary the number of estimators (trees). Importantly, we selected a subset of 273 genes from the Alzheimer's disease pathway and kept all samples.

We kept all the non-specified hyperparameters to their default values. Our implementation is based on Python 3.7.6 and the library `missingpy`. We ran the algorithm with 10 concurrent jobs.

C.3 Scalability analysis for MICE

Figures C.4 and C.5 show the runtime of *a single* iteration of the MICE algorithm [192] as we vary the number of genes and samples.

We kept all the non-specified hyperparameters to their default values. Our implementation is based on Python 3.7.6 and the library `sklearn` [298], in particular `sklearn.impute.IterativeImputer`.

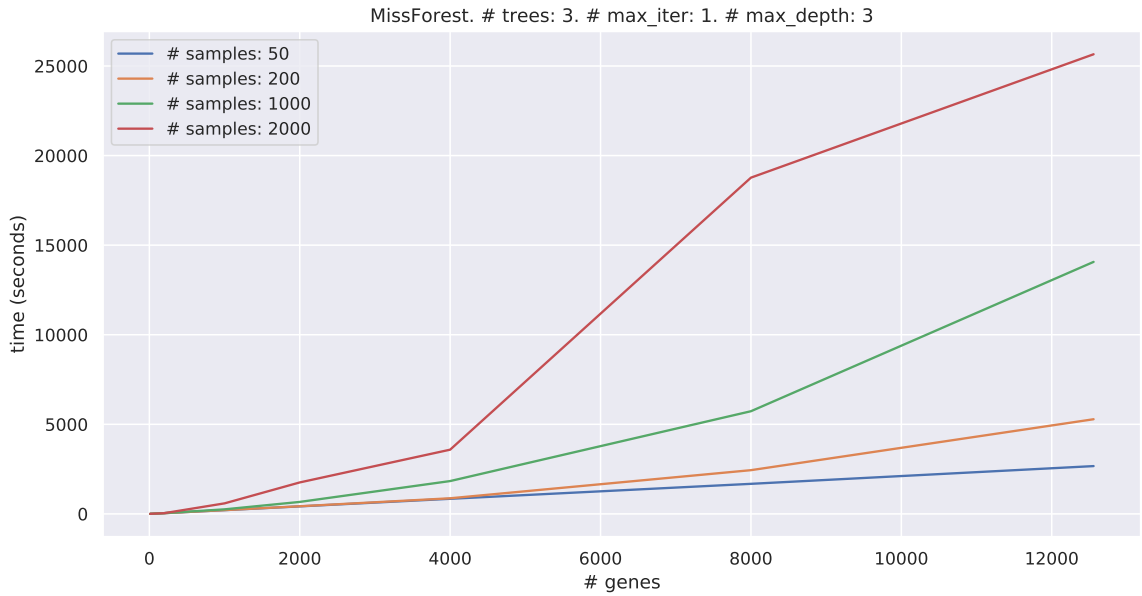


Fig. C.1 Runtime of *a single* iteration of the MissForest algorithm [193] as we vary the number of samples.

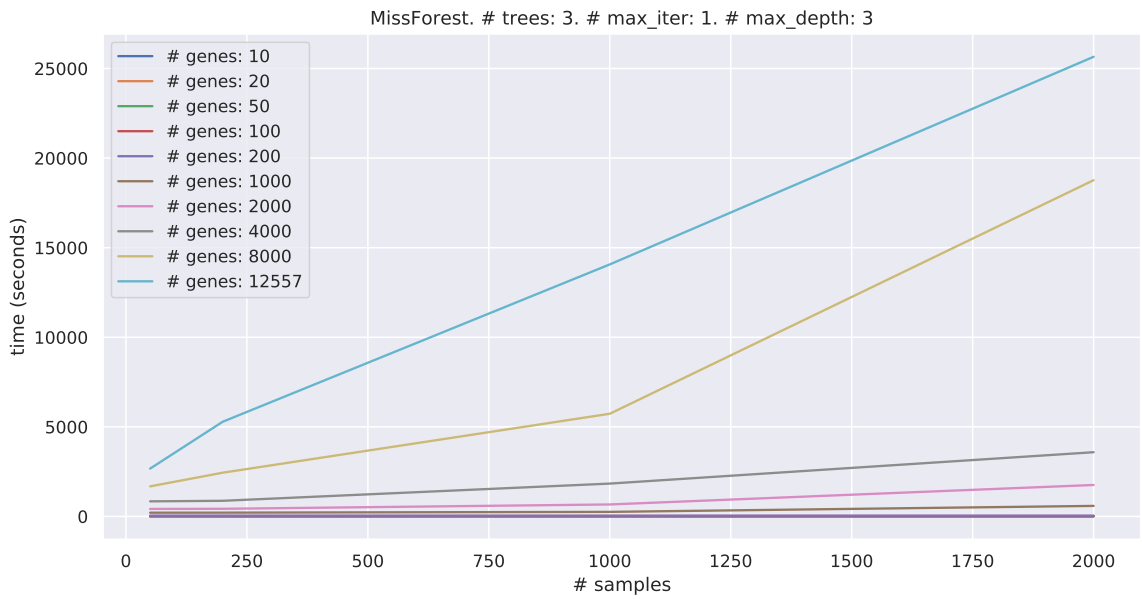


Fig. C.2 Runtime of *a single* iteration of the MissForest algorithm [193] as we vary the number of genes.

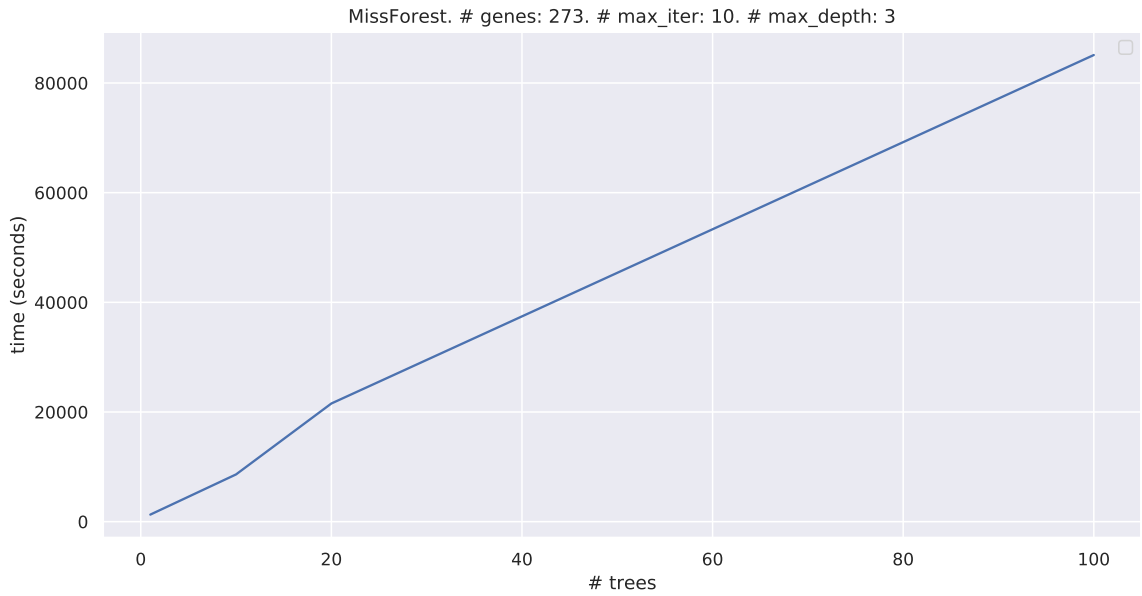


Fig. C.3 Runtime of MissForest algorithm [193] as we vary the number of trees. We ran the algorithm using all the samples on a subset of 273 trees from the Alzheimer's disease pathway.

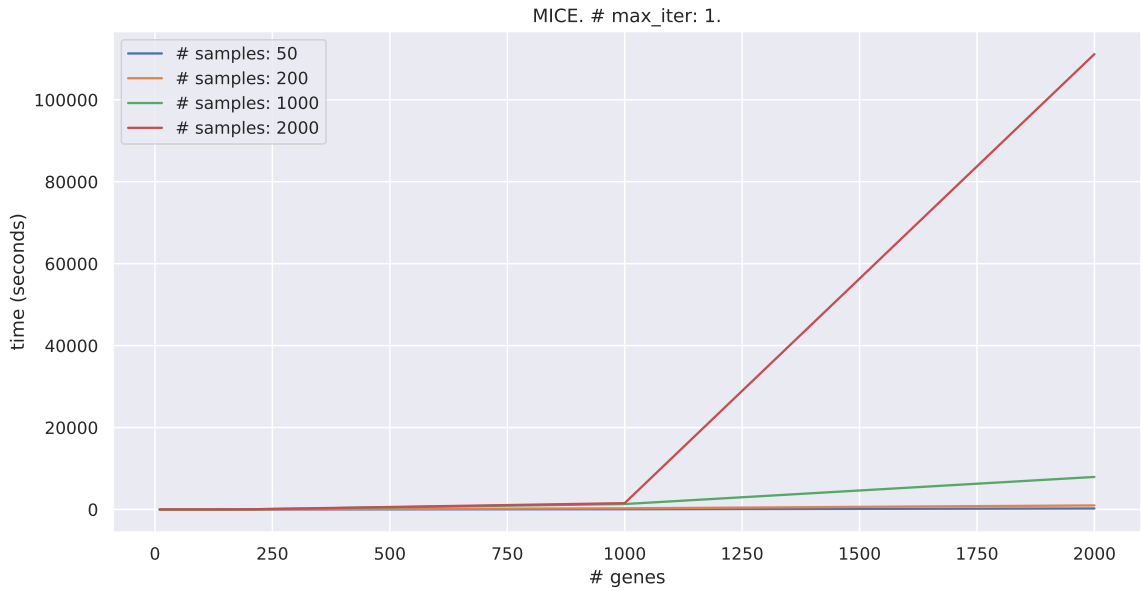


Fig. C.4 Runtime of *a single* iteration of the MICE algorithm [192] as we vary the number of genes.

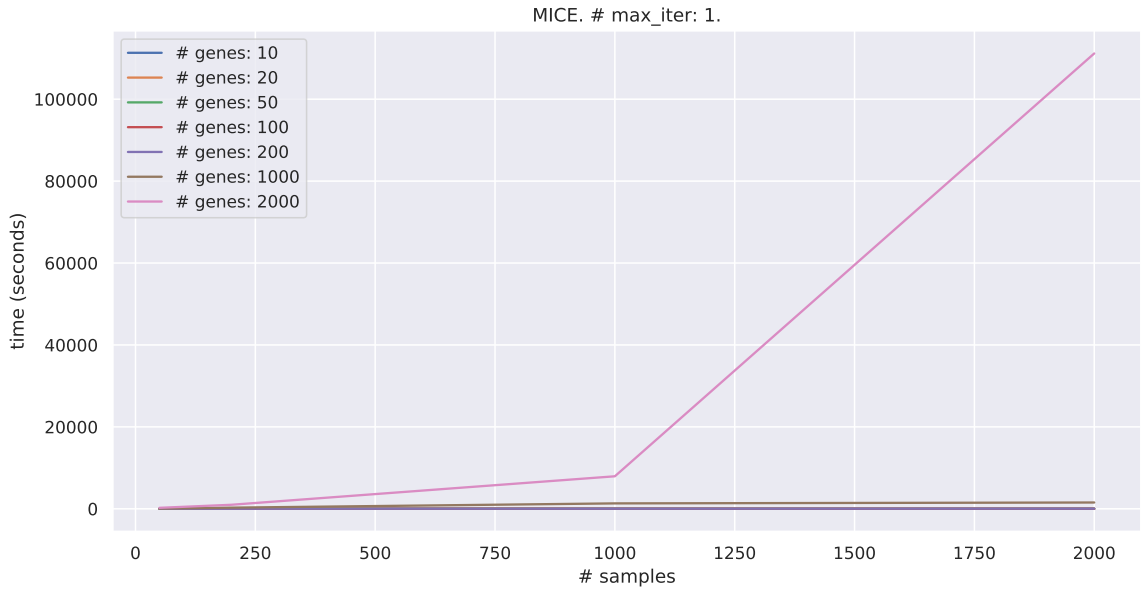


Fig. C.5 Runtime of *a single* iteration of the MICE algorithm [192] as we vary the number of samples.

	All genes		**Alzheimer**	
PMI	In-place	Inductive	In-place	Inductive
Alpha α	0.5	0.5	0.6	0.5
Beta β	0.9	0.5	0.9	0.5
Learning rate	0.0001	0.0001	0.001	0.001
Dropout probability	0	0.2	0	0.2
Number of layers	2	1	3	2
Hidden dimensionality per-layer	1366	3072	1383	1531

C.4 PMI hyperparameters

Figure C.6 shows the validation MSE for different configurations of hyperparameters of PMI. We optimise the model using `wandb` [299]. We report the selected hyperparameters for each scenario in the following table:

C.5 GAIN hyperparameters

Figure C.7 shows the validation MSE for different configurations of hyperparameters of GAIN-GTEx. We optimise the model using `wandb` [299]. We report the selected hyperparameters for each scenario in the following table:

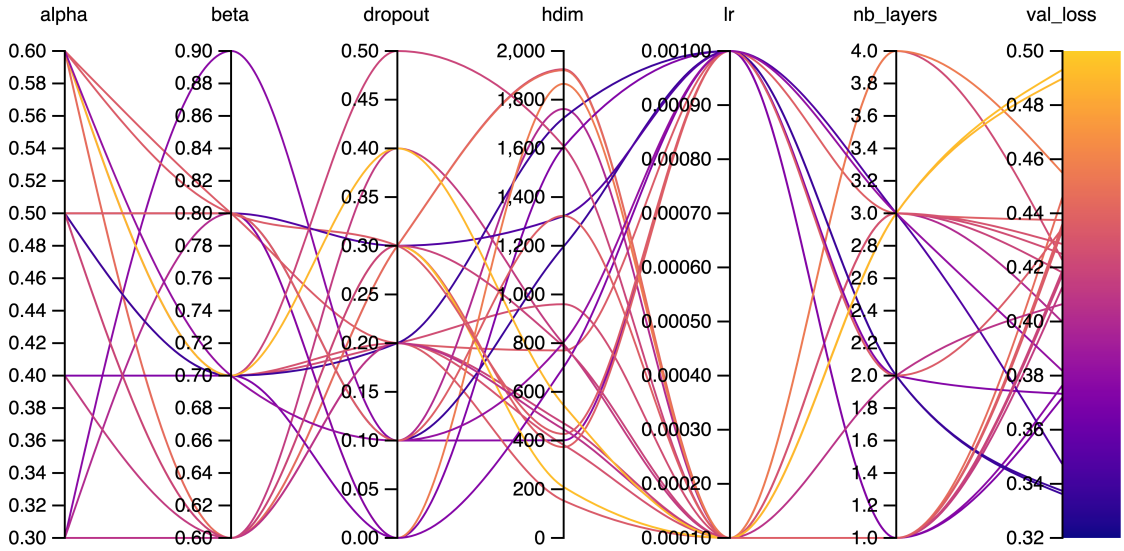


Fig. C.6 Exploration of the hyperparameter space for PIM on the subset of genes from the Alzheimer's disease pathway (*in-place mode*). The score axis shows the mean squared error on an independent validation set.

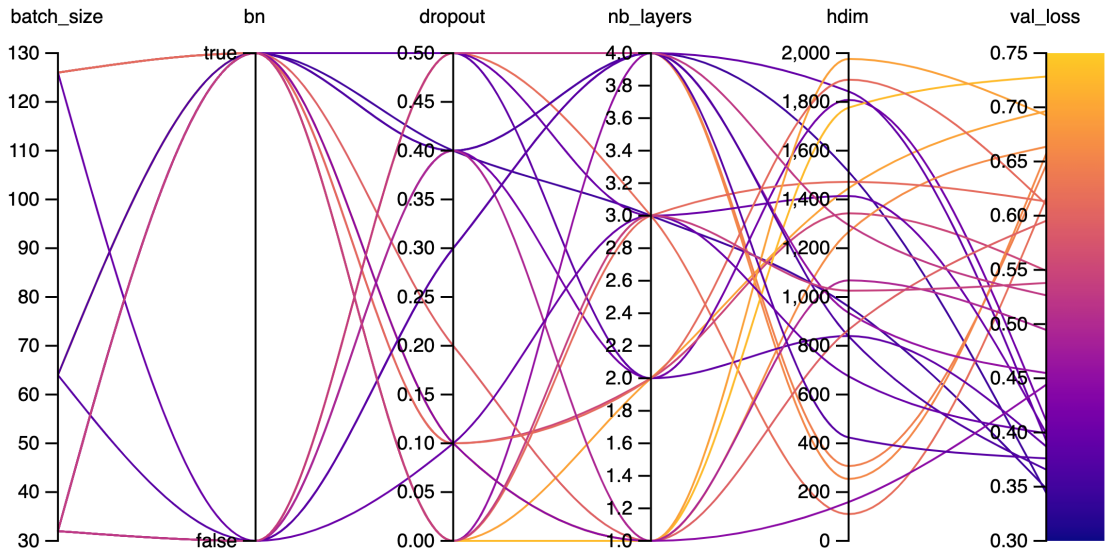


Fig. C.7 Exploration of the hyperparameter space for GTEx-GAIN on the subset of genes from the Alzheimer's disease pathway (*inductive mode*). The score axis shows the mean squared error on an independent validation set. In our experimentation we note that the model is fairly sensitive to the dimensionality of the hidden layers. On one hand, a small value leads to underfitting. On the other hand, a large value allows the model to trivially *copy-paste* the expression of the observed components.

	All genes		Alzheimer	
GAIN-GTEx	In-place	Inductive	In-place	Inductive
Learning rate	0.001	0.001	0.001	0.001
Dropout probability	0	0.2	0.4	0.4
Number of layers	1	4	3	3
Hidden dimensionality per layer	2403	1902	296	296

Regarding the output activation of GAIN, we leverage a linear and a sigmoid activation functions for the generator and discriminator, respectively. The linear activation ensures that the range of the output expression is unrestricted. We model both the generator and discriminator as MLPs with 4 hidden layers (2403 units each). In terms of the hyperparameter λ to trade off the adversarial and reconstruction losses of the generator, we find that setting $\lambda = 1$ yields good results in all settings.

Mask and hint generation. At training time, for each training example, we sample the mask vector $\mathbf{m}$ from a Bernoulli distribution $B(1, p)$ parameterised by a random probability $p = 0.5$. To generate the hint vector $\mathbf{h}$, we sample $\mathbf{b}$ from $B(1, p)$, where $p = 0.5$.

C.6 Supplementary figures

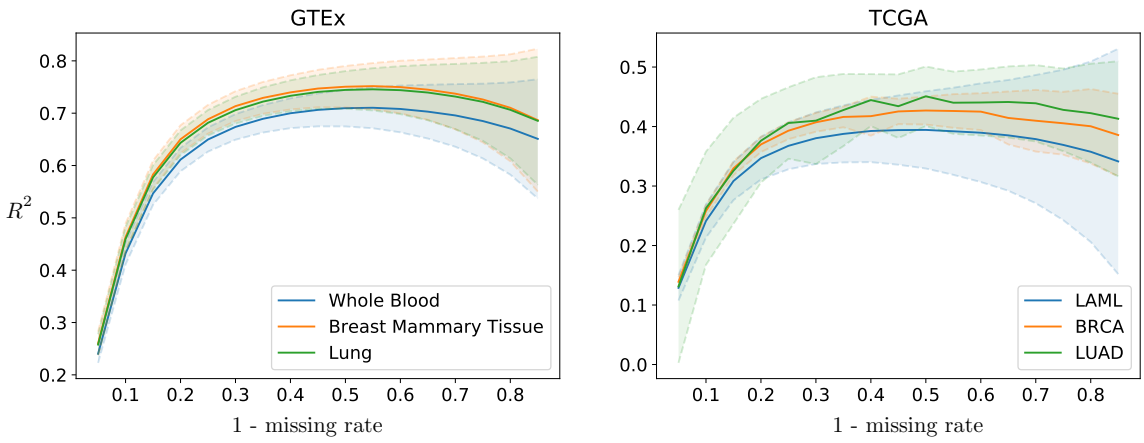


Fig. C.8 PMI R^2 imputation scores per tissue across missing rate for 3 TCGA cancer types and their healthy counterpart in GTEx. The shaded area represents one standard deviation of the per-gene R^2 scores in the corresponding tissue. The greater the rate of missingness, the lower the performance.

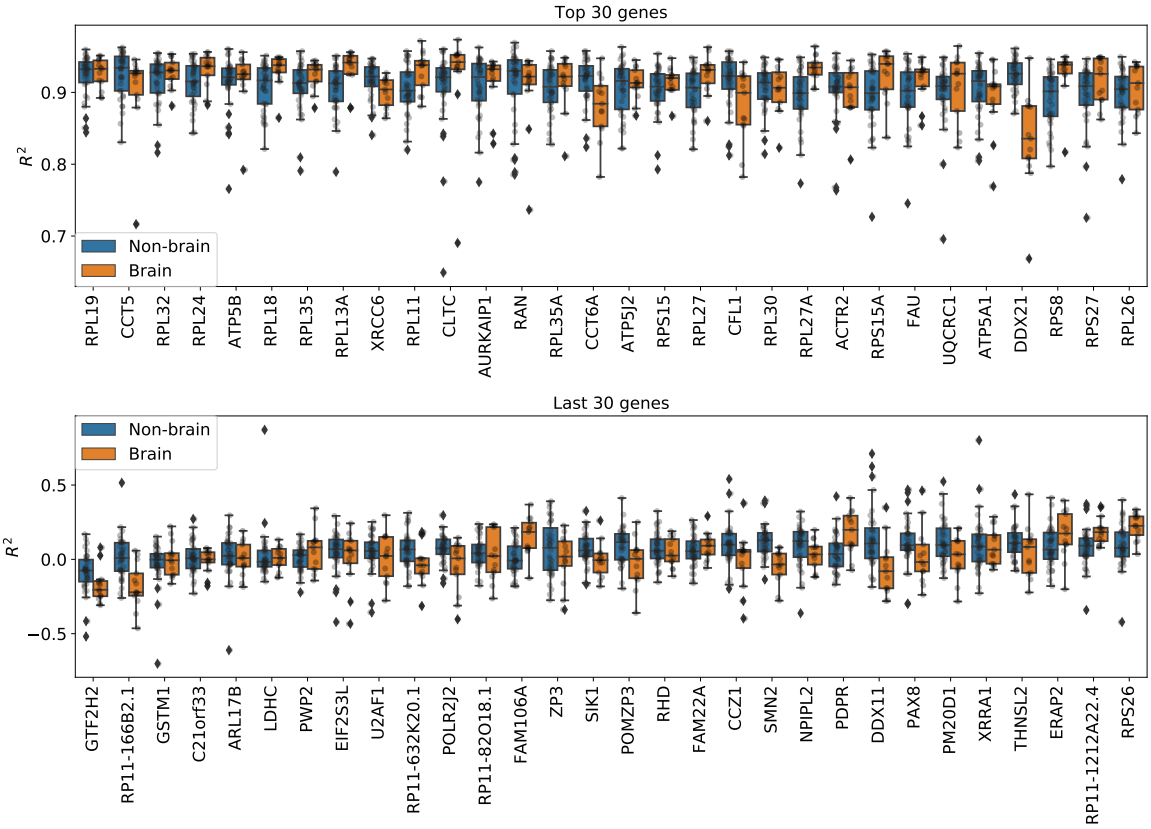


Fig. C.9 Per-gene imputation R^2 scores. We rank all the genes according to the average R^2 imputation scores across tissue types. We select the top 30 and last 30 genes. Interestingly, most of the best imputed genes are RPLs (L ribosomal proteins), which are known to be well conserved both evolutionarily and across tissue types.

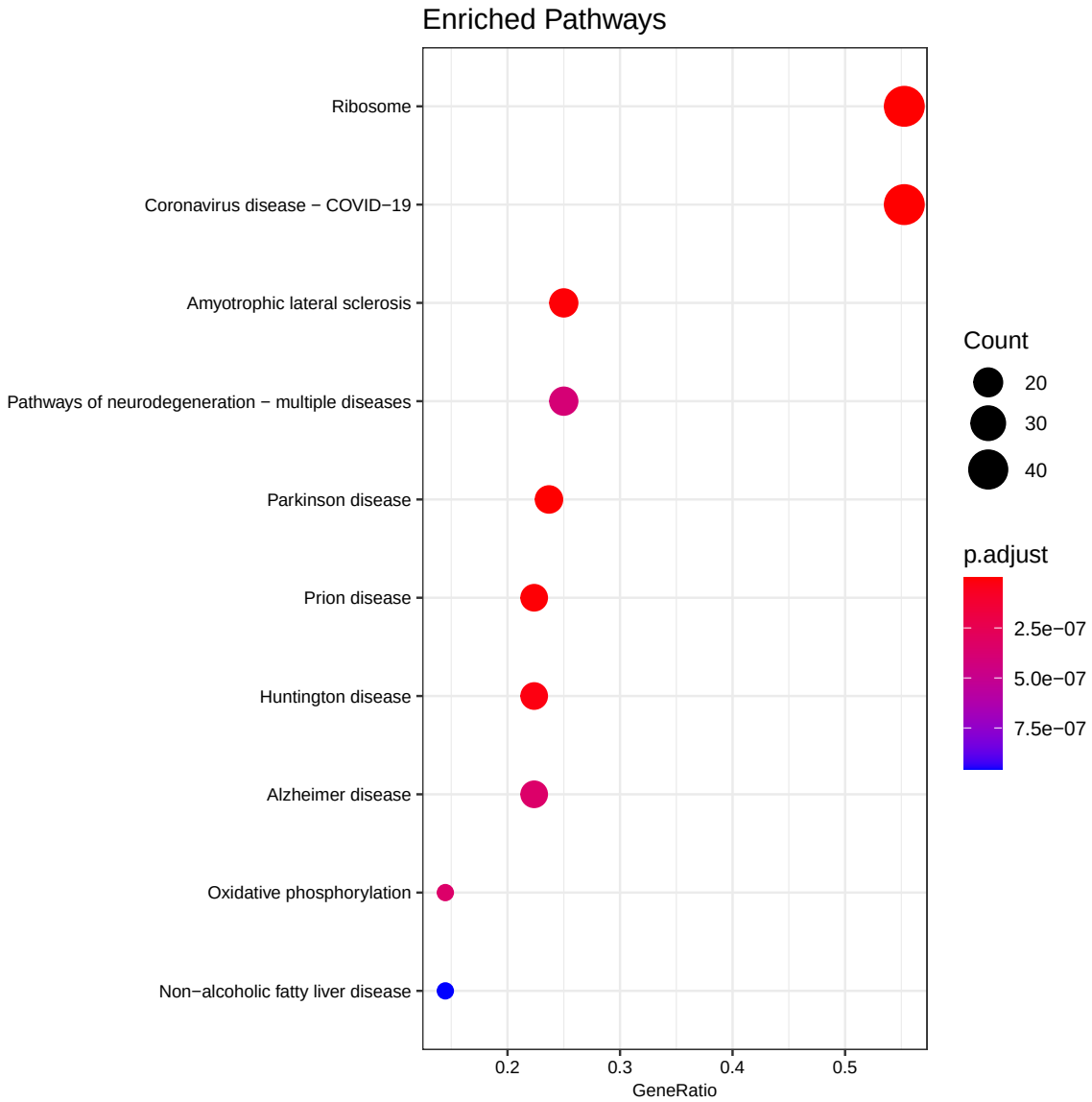


Fig. C.10 Top enriched KEGG pathways for over-representation analysis of the top 100 best-imputed genes. Interestingly, we note that most of the best-imputed genes are RPLs (L ribosomal proteins), which are generally well-conserved evolutionarily and across tissue types.

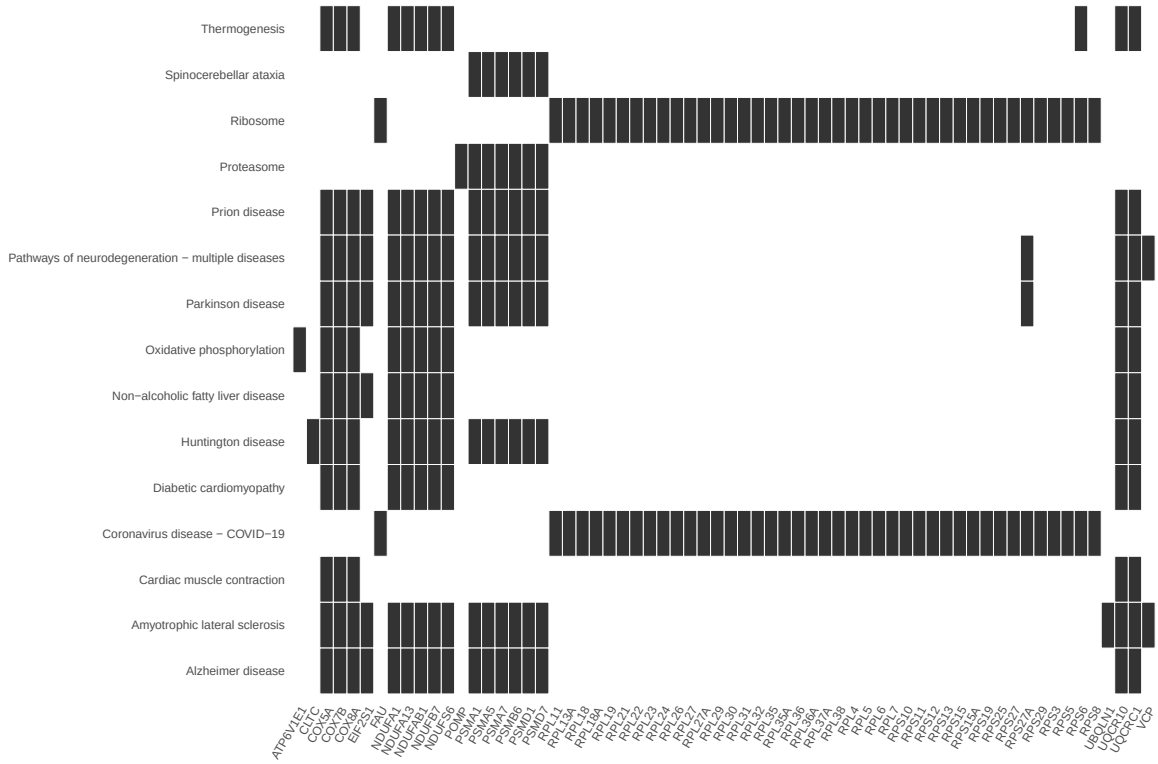


Fig. C.11 Heatmap of the gene-pathway associations for the top 100 imputed genes and the enriched KEGG pathways. Interestingly, we note that most of the best-imputed genes are RPLs (L ribosomal proteins), which are generally well-conserved evolutionarily and across tissue types.

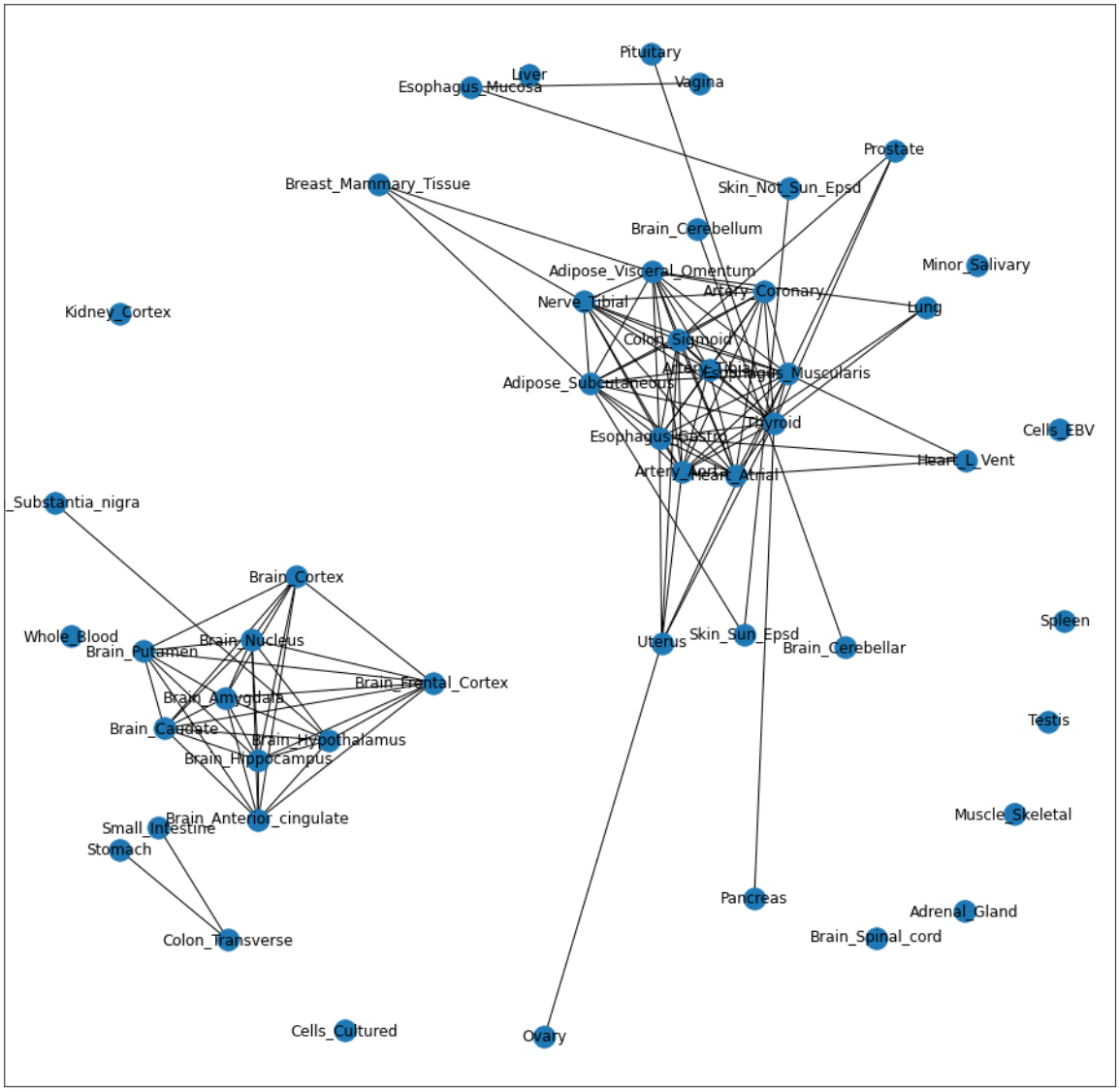


Fig. C.12 Network generated from the per-tissue R^2 scores (PMI; Alzheimer Pathway). For each pair of tissue types, we compute the Pearson's correlation coefficient between the tissue-specific vectors of per-gene R^2 scores. We then filter out the edges whose correlation is lower than an arbitrary threshold. This plot shows that the R^2 scores carry information about the tissue type and that the same genes in similar tissue types have similar R^2 scores.

Supplementary Information D

Multi-tissue imputation of gene expression

D.1 HYFA's computational complexity

Let N be the number of individuals, T the total number of tissues, and M the number of metagenes. If we consider a 3-uniform hypergraph of individuals, tissues, and metagenes, the number of nodes is $\mathcal{O}(N + T + M)$ and the number of hyperedges is $\mathcal{O}(N * T * M)$. The time complexity of every step of HYFA's message passing computation (Methods) for a single head is:

- Message computation: $\mathcal{O}((N * T * M) * d' * d)$
- Attention mechanism (assuming hidden dimension d' of attention mechanism):
  - Messages to individual nodes: $\mathcal{O}(T * M * d' * d)$
  - Messages to tissue nodes (optional): $\mathcal{O}(N * M * d' * d)$
  - Messages to metagene nodes (optional): $\mathcal{O}(N * T * d' * d)$
- Message aggregation: $\mathcal{O}((N * T * M) * d')$
- Updating node features: $\mathcal{O}((N + T + M) * d' * d)$

where d is the number of input features and d' is the number of output features. As a result, the time complexity of a single hypergraph layer is $\mathcal{O}((N * T * M + N + T + D) * d' * d)$.

D.2 Ablation of architecture

We ablate the impact of two key architectural components of HYFA: (1) representating multi-tissue gene expression as a hypergraph of individuals, metagenes, and tissues; and (2) the design of a specialised hypergraph message passing neural network layer with attentional aggregation.

Number of metagenes In Supplementary Figure D.1, we plot the validation loss and correlation coefficient vs. the number of metagenes for both attentional (GAT) and standard message passing (MPNN). The attentional model (GAT) refers to HYFA with an attention-based aggregation mechanism (Chapter 5), while the message passing model (MPNN) refers to HYFA with simple average aggregation (i.e. mean across all incoming messages). For each number of metagenes, we ran hyperparameter optimisation with wandb [299] to obtain the loss and Pearson correlation coefficient ρ for the best performing model (we ran sweeps with a maximum of 100 runs). For fair comparison across runs, validation metrics were computed for a fixed subset of target tissues: 'Lung', 'Pancreas', 'Heart_Atrial', and 'Esophagus_Muscularis'. The hyperparameter values considered for ablation studies are available in Supplementary Table D.3.

As noted in Chapter 5, modulating the number of metagenes controls the growth of the receptive field for each node in the hypergraph and helps alleviate over-squashing. Setting very low number of metagenes is computationally fast during training and inference but may compress fine-grained information (e.g. setting metagenes to 1 results in a bipartite graph of individuals and tissues), while a very high number of metagenes preserves fine-grained relationships between genes, tissues, and individuals but may become computationally intractable. Supplementary Figure D.1 shows that there is a 'sweet spot' for the number of metagenes between 50-100 that leads to optimal performance. Additionally, as shown in Supplementary Figure D.1d, using 200 or more metagenes can consume upwards of 20 GB of GPU memory (or more, depending on other hyperparameters), which makes training and hyperparameter tuning expensive/intractable on academic GPUs.

For our best performing model using 50 metagenes, the average iteration time to perform a forward pass for the optimal minibatch size of 63 is 119.72 ms during training and 61.70 ms during inference. The average GPU usage for the same are 6.8 GB and 3.3 GB, respectively. Metrics are computed on a single NVIDIA RTX 8000 GPU (48 GB) and 16 core CPU, averaged across 80 minibatches per model.

Hypergraph message passing architecture Supplementary Table D.1 summarises results for the best performing GAT and MPNN, demonstrating that the specialised hypergraph attentional aggregation brings notable gains in imputation performance. This is consistent with the observation that, through the attention mechanism, the model can prioritise certain messages from others, alleviating the over-squashing problem. As a naïve baseline, we also show results for a structure-agnostic model which does not perform any message passing and simply predicts the hyperedge attributes via an MLP. Supplementary Table D.2, in addition, shows an ablation of the demographic covariates. The inductive bias of reusing knowledge across tissues and metagenes via message passing seem critical for gene expression imputation.

Table D.1 Ablation study of hypergraph message passing design. A specialised hypergraph attentional aggregation brings significant gains in imputation performance over standard message passing as well as a naïve structure-agnostic baseline.

GNN Layer	Val. Loss ↓	Val. Correlation ↑
Structure-agnostic MLP	0.9719	0.0396
Message Passing (MPNN)	0.7488	0.4499
Attentional (GAT)	**0.7393**	**0.4614**

Table D.2 Ablation study of demographic covariates. Demographic covariates have a small impact on the overall validation performance.

Demographic covariates	Val. Loss ↓	Val. Correlation ↑
Demographic covariates	**0.7393**	**0.4614**
Randomly shuffled covariates	0.7479	0.4527
Without demographic covariates	0.7414	0.4587

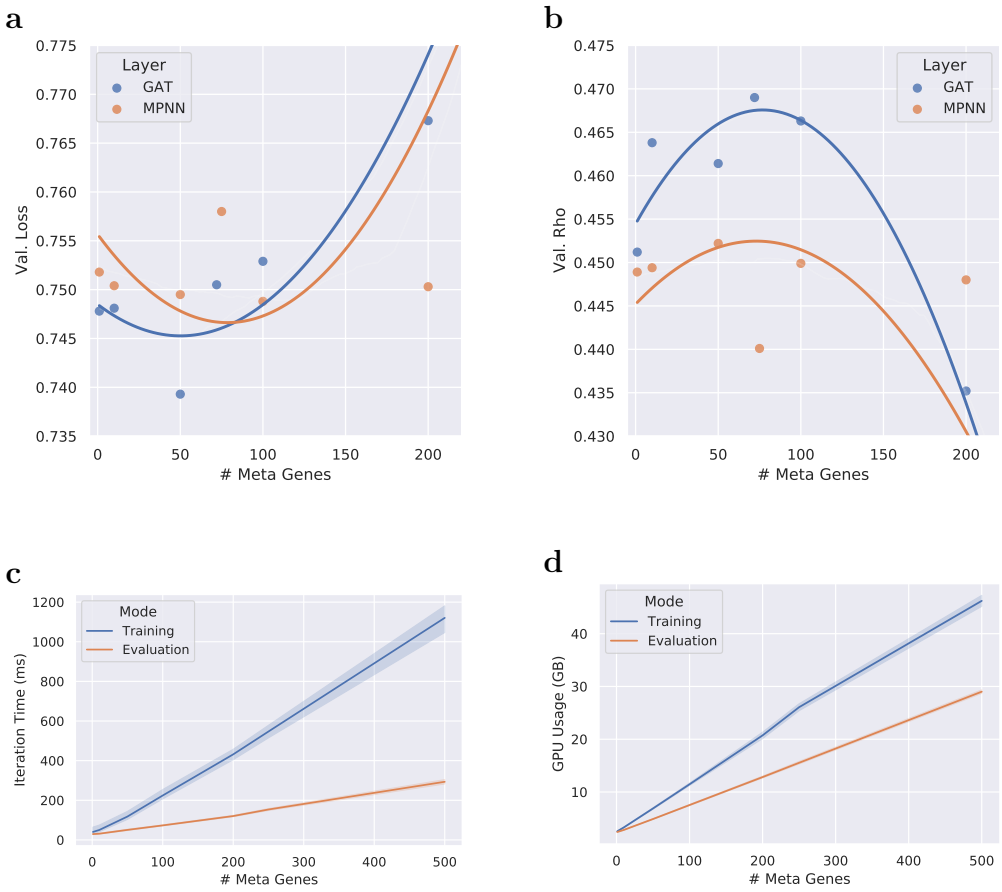


Fig. D.1 Impact of number of metagenes in hypergraph representations vs. (a, b) model performance and (c, d) scalability. (a, b) There is a 'sweet spot' for the number of metagenes between 50-100 that leads to optimal performance for both attentional (GAT) and standard message passing (MPNN). Curves are estimated via a polynomial regression with order 2. (c, d) Impact of number of metagenes in hypergraph representations vs. model scalability in terms of average minibatch iteration time and GPU usage (batch size = 63). Training and hyperparameter tuning for models with upwards of 200 metagenes becomes intractable on academic GPUs. Bands denote 99% confidence interval and the centre of the error bands corresponds to the mean. Figure credit: Chaitanya Joshi.

Table D.3 Hyperparameter values considered for ablation studies. We used wandb [299] to run Bayesian hyperparameter search over the variables and ranges considered. Note that with the Attentional GAT layer, the total dimension of the message $\boldsymbol{m}_{ijk}$ is multiplied by the number of attention heads (here, $28 \times 28 = 784$).

Hyperparameter	Values Considered	Best Value
GNN Layer	{ GAT, MPNN }	GAT
Num. Metagenes	10 – 200	50
Num. Message Passing Layers	1 – 3	2
Num. MLP Layers (within GNN)	1 – 2	1
Num. MLP Layers (Prediction head)	1 – 2	2
Num. Attention Heads (GAT only)	4 – 32	28
Dimension of Donor Emb. $\boldsymbol{h}^d$	16 – 128	71
Dimension of Metagene Emb. $\boldsymbol{h}^m$	16 – 128	48
Dimension of Tissue Emb. $\boldsymbol{h}^t$	16 – 128	120
Dimension of Hyperedge Attr. $\boldsymbol{e}_{ij}$	16 – 128	98
Dimension of Message $\boldsymbol{m}_{ijk}$	16 – 128	28
Learning Rate	0.0001 – 0.005	0.00045
Batch Size	16 – 64	63
Dropout	0.0 – 0.5	0.17385
Normalisation	{ BatchNorm, LayerNorm, None }	BatchNorm
Activation Function	{ ReLU, Swish }	Swish

D.3 Connection with maximum likelihood

Let x_{obs} be a random variable denoting the observed data (e.g. multi-tissue gene expression with missing values corresponding to uncollected tissues). Our optimisation procedure (Methods) splits x_{obs} into *pseudo-observed* $\hat{\mathrm{x}}_{obs}$ and *pseudo-missing* $\hat{\mathrm{x}}_{mis}$ values, that is, $\mathrm{x}_{obs} = (\hat{\mathrm{x}}_{obs}, \hat{\mathrm{x}}_{mis})$. The log-likelihood of the observed data then corresponds to:

$$\log p(\mathrm{x}_{obs}) = \log p(\hat{\mathrm{x}}_{obs}, \hat{\mathrm{x}}_{mis}) = \log p(\hat{\mathrm{x}}_{mis}|\hat{\mathrm{x}}_{obs}) + \log p(\hat{\mathrm{x}}_{obs})$$

and $\log p(\hat{\mathrm{x}}_{mis}|\hat{\mathrm{x}}_{obs})$ is precisely the quantity that our loss function is maximising through the pseudo-mask mechanism (Methods).

D.4 Training algorithm

Optimisation. We minimise the mean squared error $\mathcal{L}$ between the normalised, ground-truth gene expression $\boldsymbol{x}_i^{(u)}$ and the imputed values $\hat{\boldsymbol{x}}_i^{(u)}$:

$$\mathcal{L}(\boldsymbol{x}_i^{(u)}, \hat{\boldsymbol{x}}_i^{(u)}) = \frac{1}{G}\left(\boldsymbol{x}_i^{(u)} - \hat{\boldsymbol{x}}_i^{(u)}\right)^\top \left(\boldsymbol{x}_i^{(u)} - \hat{\boldsymbol{x}}_i^{(u)}\right)$$

where G is the number of genes. At train time, for any given individual, we dynamically mask out the expression values of a measured tissue type at random and treat them as uncollected, i.e. the ground truth. Algorithm 2 summarises the training algorithm.

Algorithm 2: Training algorithm

Input: Input dataset $\{\mathcal{X}(i), \mathcal{T}(i)\}_{i=1}^N$, model f
while *not convergence criteria reached* **do**
  Sample mini-batch $\mathbb{B}$ of individuals
  foreach *individual i in mini-batch $\mathbb{B}$* **do**
    Choose collected tissues $\mathcal{C}$ and uncollected tissue u:
      $u \sim \mathcal{T}(i), \quad \mathcal{C} = \mathcal{T}(i) - \{u\}$
    Predict gene expression of proxy uncollected tissue u:
      $\hat{\boldsymbol{x}}_i^{(u)} = f(u, \{\boldsymbol{x}_i^{(k)} | k \in \mathcal{C}\})$
  end
  Optimise the model by descending its stochastic gradient:
    $\nabla \frac{1}{|\mathbb{B}|} \sum_{i \in \mathbb{B}} \mathcal{L}(\boldsymbol{x}_i^{(u)}, \hat{\boldsymbol{x}}_i^{(u)})$
end

D.5 Training HYFA via variational inference

HYFA can alternatively be trained via variational inference by introducing a variational distribution $q(\mathbf{Z}|\tilde{\mathbf{X}}, \mathbf{U}) = \prod_i^N q(\boldsymbol{z}_i|\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i)$, where $\boldsymbol{z}_i$ is a latent variable that explains the high-dimensional, multi-tissue gene expression data.

Parameters of inference model Given the updated donor representations $\hat{\boldsymbol{h}}_i^p$, we compute the parameters of the inference model $q(\boldsymbol{z}_i|\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i) = \mathcal{N}(\boldsymbol{z}_i; \boldsymbol{\mu}_i, \mathrm{diag}(\boldsymbol{\sigma}_i^2))$ as follows:

$$\boldsymbol{\mu}_i = f_\mu(\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i; \boldsymbol{\phi}) = \mathrm{MLP}(\hat{\boldsymbol{h}}_i^p) \qquad \log \boldsymbol{\sigma}_i = f_\sigma(\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i; \boldsymbol{\phi}) = \mathrm{MLP}(\hat{\boldsymbol{h}}_i^p),$$

where MLP denotes a multilayer perceptron.

Parameters of generative model Assuming a Gaussian likelihood, for a given sample $\boldsymbol{z}_i \sim q(\boldsymbol{z}_i|\tilde{\boldsymbol{X}}_i, \boldsymbol{u}_i)$, we compute the parameters of the generative model $p(\boldsymbol{x}_i^{(k)}|\boldsymbol{z}_i, \boldsymbol{u}_i, k)$ as follows:

$$p(\boldsymbol{x}_i^{(k)}|\boldsymbol{z}_i, \boldsymbol{u}_i, k) = \prod_j^G p(x_{ij}^{(k)}|\boldsymbol{z}_i, \boldsymbol{u}_i, j, k) \qquad p(x_{ij}^{(k)}|\boldsymbol{z}_i, \boldsymbol{u}_i, j, k) = \mathcal{N}(x_{ij}^{(k)}; \mu_{ij}^{(k)}, \sigma^{2(k)}_{ij}),$$

where the mean $\mu_{ij}^{(k)}$ and standard deviation $\sigma_{ij}^{(k)}$ are computed as follows:

$$\begin{aligned}
\boldsymbol{\mu}_i^{(k)} &= \boldsymbol{W}_\mu \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\mu \\
\boldsymbol{\sigma}_i^{(k)} &= \mathrm{softplus}\Big(\boldsymbol{W}_\sigma \hat{\boldsymbol{e}}_i^{(k)} + \boldsymbol{b}_\sigma\Big) \\
\hat{\boldsymbol{e}}_i^{(k)} &= \mathrm{MLP}\Big(\Big\|_{j=1}^M \hat{\boldsymbol{e}}_{ij}^{(k)}\Big) \\
\hat{\boldsymbol{e}}_{ij}^{(k)} &= \mathrm{MLP}(\boldsymbol{z}_i, \boldsymbol{h}_j^m, \boldsymbol{h}_k^t),
\end{aligned}$$

where $\boldsymbol{W}_\mu$, $\boldsymbol{W}_\sigma$, $\boldsymbol{b}_\mu$, and $\boldsymbol{b}_\sigma$ are learnable parameters and $\mathrm{softplus}(x) = \log\Big(1+\exp(x)\Big)$.

Optimisation We maximise the evidence lower bound on the data log-likelihood:

$$\log p(\tilde{\mathbf{X}}|\mathbf{U}) \geq \mathbb{E}_{q(\mathbf{Z}|\tilde{\mathbf{X}},\mathbf{U})}[\log p(\tilde{\mathbf{X}}|\mathbf{Z}, \mathbf{U}) + \log p(\mathbf{Z}|\mathbf{U}) - \log q(\mathbf{Z}|\tilde{\mathbf{X}}, \mathbf{U})]$$

where the prior $p(\mathbf{Z}|\mathbf{U})$ is a factorised normal distribution conditioned on demographic information:

$$p(\mathbf{Z}|\mathbf{U}) = \prod_{i}^{N} p(\boldsymbol{z}_i|\boldsymbol{u}_i) \qquad p(\boldsymbol{z}_i|\boldsymbol{u}_i) = \mathcal{N}(\boldsymbol{z}_i; \boldsymbol{\mu}'_i, \mathrm{diag}(\boldsymbol{\sigma}'^2_i)),$$

with parameters $\boldsymbol{\mu}'_i = \mathrm{MLP}_{\mu'}(\boldsymbol{u}_i)$ and $\log \boldsymbol{\sigma}'_i = \mathrm{MLP}_{\sigma'}(\boldsymbol{u}_i)$. Importantly, leveraging a factorised prior conditioned on auxiliary variables guarantees identifiability under certain conditions [300].

Inference of uncollected gene expression measurements We infer the gene expression values $\hat{\boldsymbol{x}}_i^{(v)}$ of an uncollected tissue v from a given donor i as follows:

$$\hat{x}_{ij}^{(v)} = \mathbb{E}_{q(\boldsymbol{z}_i|\tilde{\boldsymbol{X}}_i,\boldsymbol{u}_i)}\left[\mathbb{E}_{p(x_{ij}^{(v)}|\boldsymbol{z}_i,\boldsymbol{u}_i,j,v)}\left[x_{ij}^{(v)}\right]\right]$$

In other words, given the multi-tissue gene expression $\tilde{\boldsymbol{X}}_i$ and demographic information $\boldsymbol{u}_i$, we compute the expectation of the target gene expression $\hat{\boldsymbol{x}}_i^{(v)}$ over the inference and generative models.

D.6 Data missingness assumption

By employing maximum likelihood inference on the observed data (Supplementary Information D.3), HYFA assumes that the data (i.e. tissues) are *Missing At Random* (MAR; [189]), that is, the missingness mechanism is independent of the unobserved data. Training HYFA via variational inference (Supplementary Information D.5) also necessitates the MAR assumption which, similar to [301], arises from maximising the log-likelihood of the observed data through the Evidence Lower Bound (ELBO). The MAR assumption is less restrictive than the *Missing At Completely at Random* (MCAR) assumption — the missingness pattern is independent of the observed and unobserved data — of other methods such as mean imputation and GAIN [184].

HYFA does not support data Missing Not At Random (MNAR), where the missingness mechanism depends on the unobserved data, i.e. the probability of being missing depends on unknown reasons [302]. To handle this scenario, we would need to model the joint distribution $p(\tilde{\boldsymbol{X}}, \boldsymbol{R})$ of the observed data $\tilde{\boldsymbol{X}}$ and the missingness mechanism $\boldsymbol{R}$, that is, the missingness mechanism would be nonignorable and would need to be explicitly modelled. This could be achieved through selection modeling [303], which factorises the joint distribution as $p(\tilde{\boldsymbol{X}}, \boldsymbol{R}) = p(\boldsymbol{R}|\tilde{\boldsymbol{X}})p(\tilde{\boldsymbol{X}})$, or pattern-mixture

models [304], which decompose the joint as $p(\tilde{\boldsymbol{X}}, \boldsymbol{R}) = p(\tilde{\boldsymbol{X}})p(\tilde{\boldsymbol{X}}|\boldsymbol{R})$. In general, it is impossible to test if MAR holds in a dataset [305], but the impact of incorrectly assuming MAR is often minor [306].

D.7 GTEx statistics

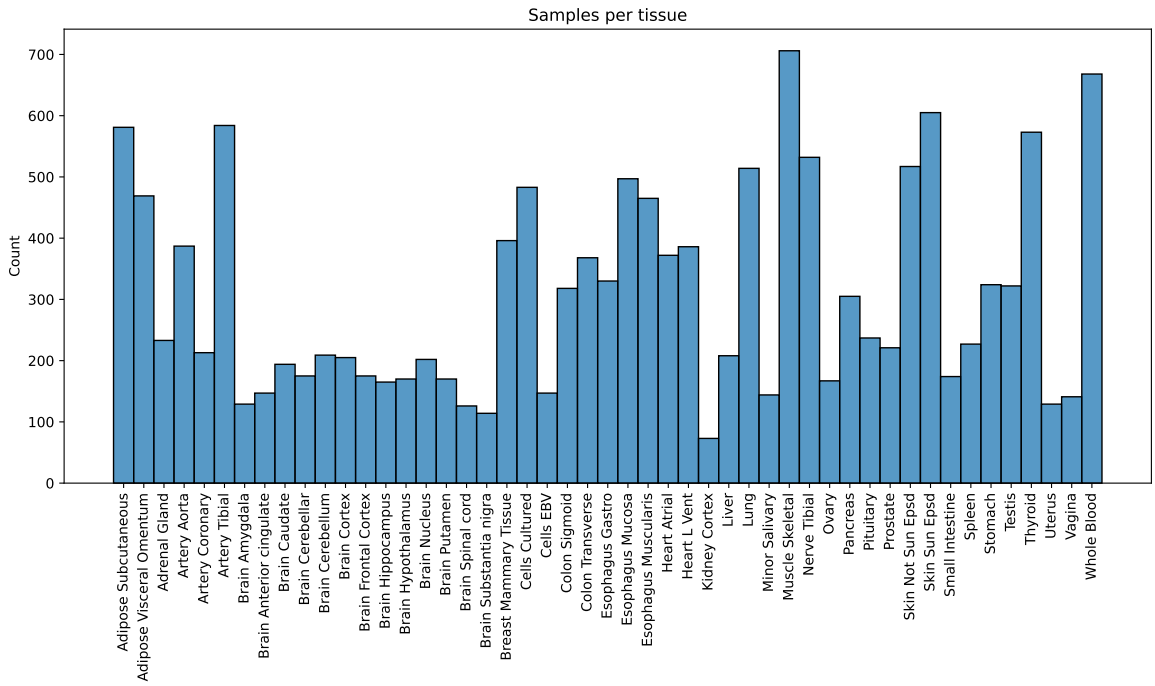


Fig. D.2 Number of samples per tissue

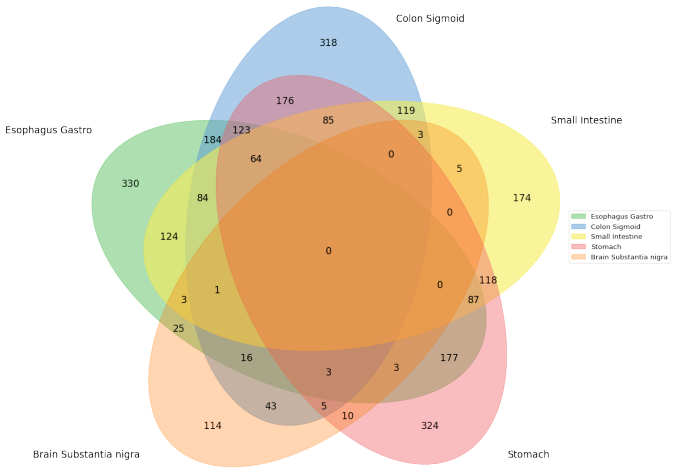
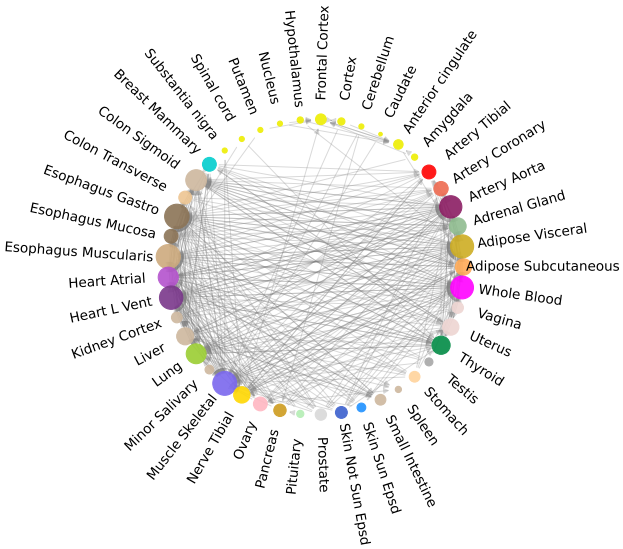


Fig. D.3 Donor overlap between brain and gastrointestinal tissues.

D.8 Per-gene prediction scores

a



b

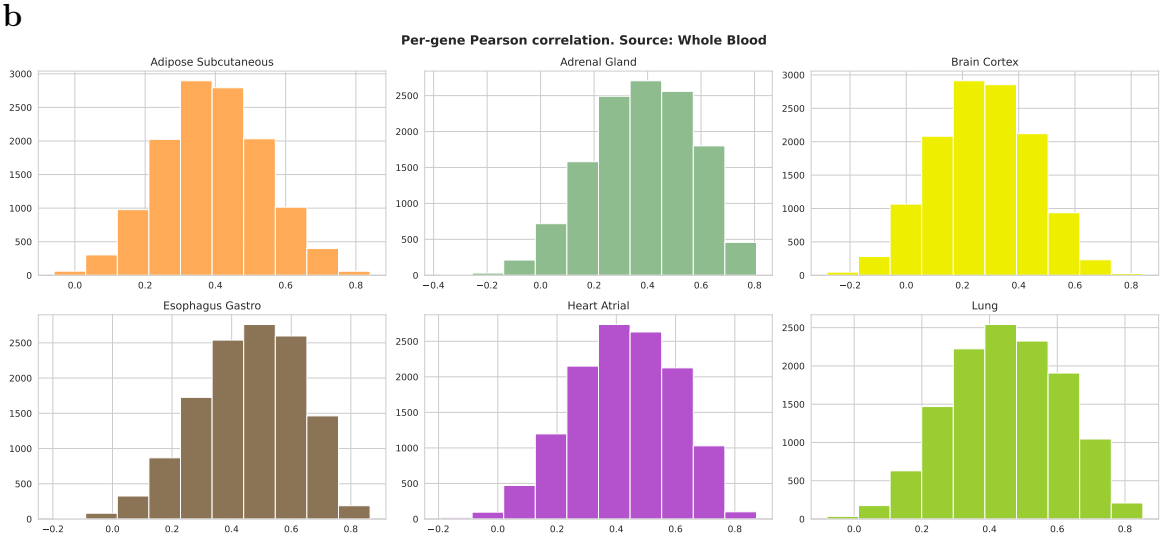


Fig. D.4 Summary of per-gene prediction scores. (a) Network of tissues depicting the predictability of target tissues with HYFA using the average per-gene Pearson ρ correlation coefficients. Edges from reference to target tissues indicate an average per-gene $\rho > 0.4$. The dimension of each node is proportional to its degree. (b) Distribution of per-gene Pearson correlation coefficients in 6 target tissues (source tissue: whole blood). We attribute the unimodality of the distributions to the fact that the data was inverse Normal transformed (Methods).

D.9 Whole blood to lung predictions

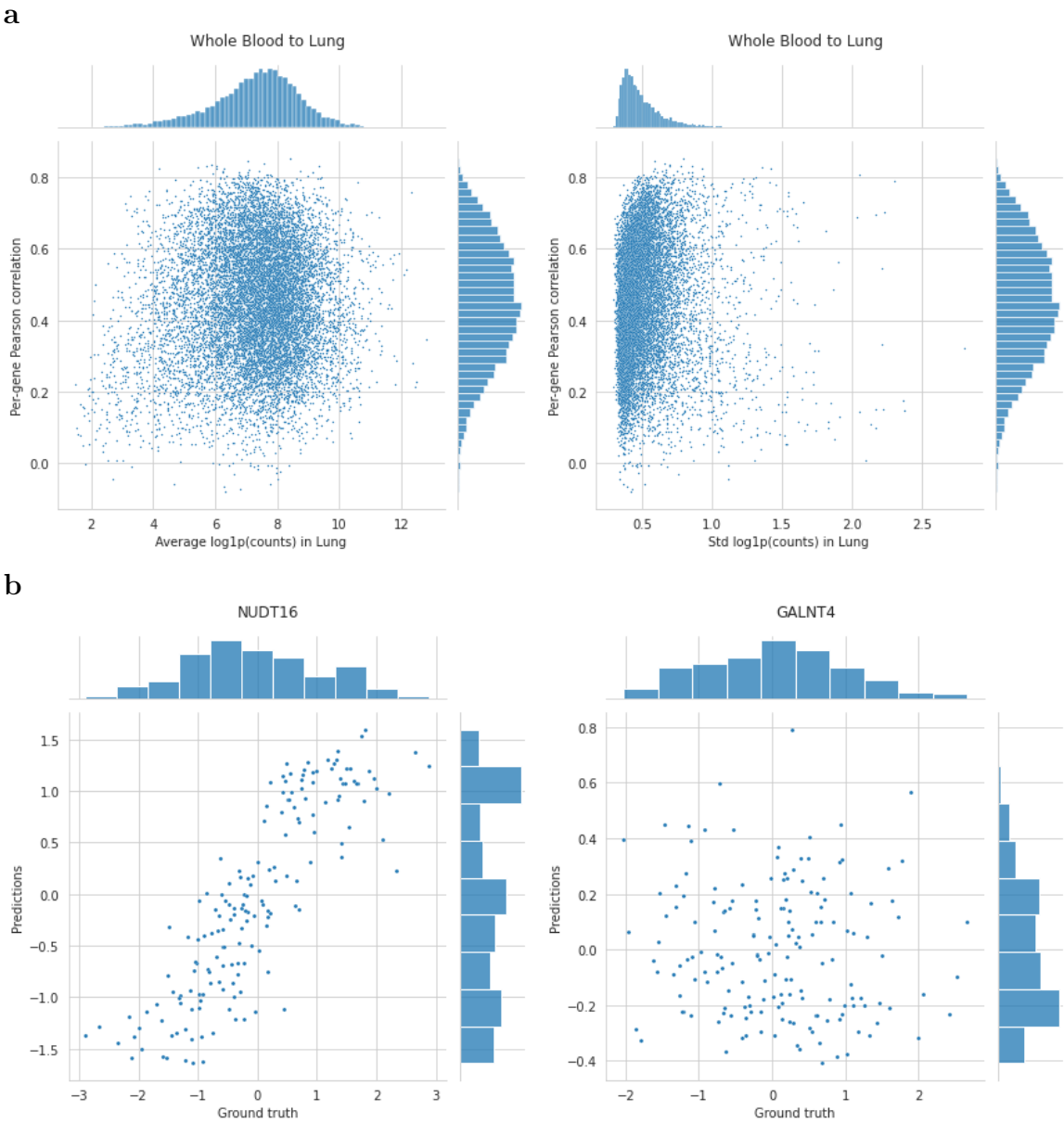


Fig. D.5 Whole blood to lung predictions for unseen individuals. (a) Average and standard deviation of per-gene expression in lung versus prediction performance (prediction performance (Pearson correlation between predicted and ground truth expression; whole blood to lung). The per-gene predictions were uncorrelated with the averages and variances of the per-gene expression in the target tissue (average: $\rho = 0.07$, variance: $\rho = 0.06$). (b) Best and worst predicted lung genes (*NUDT16*: $\rho = 0.85$; *GALNT4*: $\rho = -0.08$; n=166).

D.10 Prediction scores on Alzheimer's disease genes

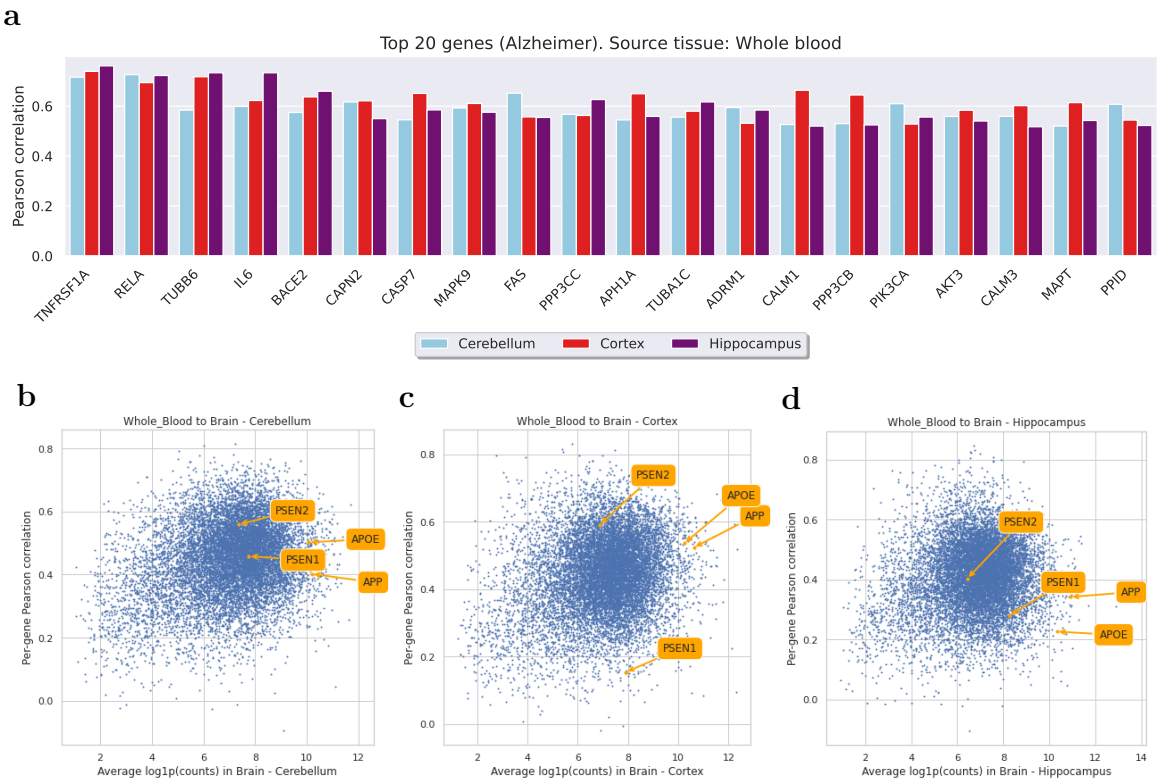


Fig. D.6 Top predicted Alzheimer's disease-relevant genes in multiple brain regions, with whole blood as reference tissue. (a) Pearson correlation coefficient of top 20 predicted genes from the Alzheimer's disease pathway (KEGG), ranked by average correlation. (b, c, d) Average per-gene expression (x-axis) versus prediction performance (Pearson correlation between predicted and ground truth expression) in (b) cerebellum, (c) cortex, and (d) hippocampus. HYFA exhibits strong prediction performance for several Alzheimer's disease-relevant genes including *APOE* (cortex $\rho = 0.536$, cerebellum: $\rho = 0.502$), *APP* (cortex $\rho = 0.524$), *PSEN1* (cerebellum: $\rho = 0.459$), and *PSEN2* (cortex: $\rho = 0.590$, cerebellum: $\rho = 559$, hippocampus: $\rho = 0.403$). In cerebellum, *PSEN1* ($\rho = 0.459$), *PSEN2* ($\rho = 0.559$), and *APOE* ($\rho = 0.502$) attained above expected performances (average $\rho = 0.448$). *APP* ($\rho = 0.524$), *PSEN2* ($\rho = 0.590$), and *APOE* ($\rho = 0.536$) surpassed the expected correlation in cortex (average $\rho = 0.443$).

D.11 Prediction scores for different accessible tissues as reference

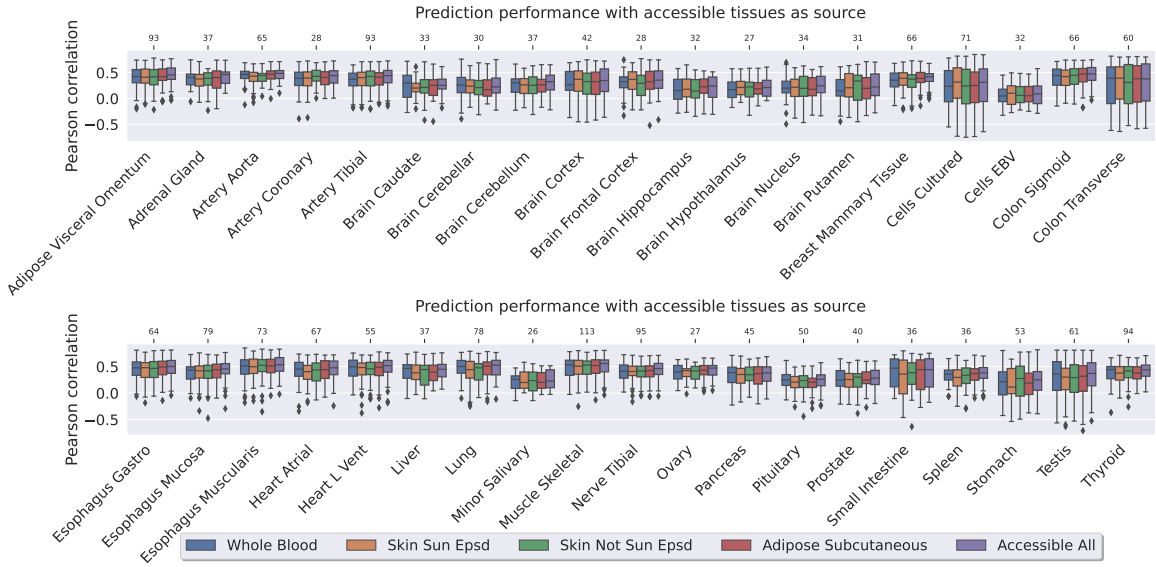


Fig. D.7 Prediction scores for different accessible tissues as reference. For each target tissue, we predicted the expression values based on accessible tissues (whole blood, skin sun exposed, skin not sun exposed, and adipose subcutaneous). We report the Pearson correlation coefficient between the predicted values and the actual gene expression values. For any given target tissue, we used the same set of individuals to evaluate performance, namely individuals in the validation and test sets with collected gene expression measurements in all the corresponding tissues. Target tissues represented by less than 25 test individuals were discarded. HYFA attains the best performance in 32 out of 38 tissues when all accessible tissues are simultaneously used as reference. Boxes show quartiles, centerlines correspond to the median, and whiskers depict the distribution range (1.5 times the interquartile range). Outliers outside of the whiskers are shown as distinct points. The top axis indicates the total number of samples for every target tissue.

D.12 Per-gene prediction scores

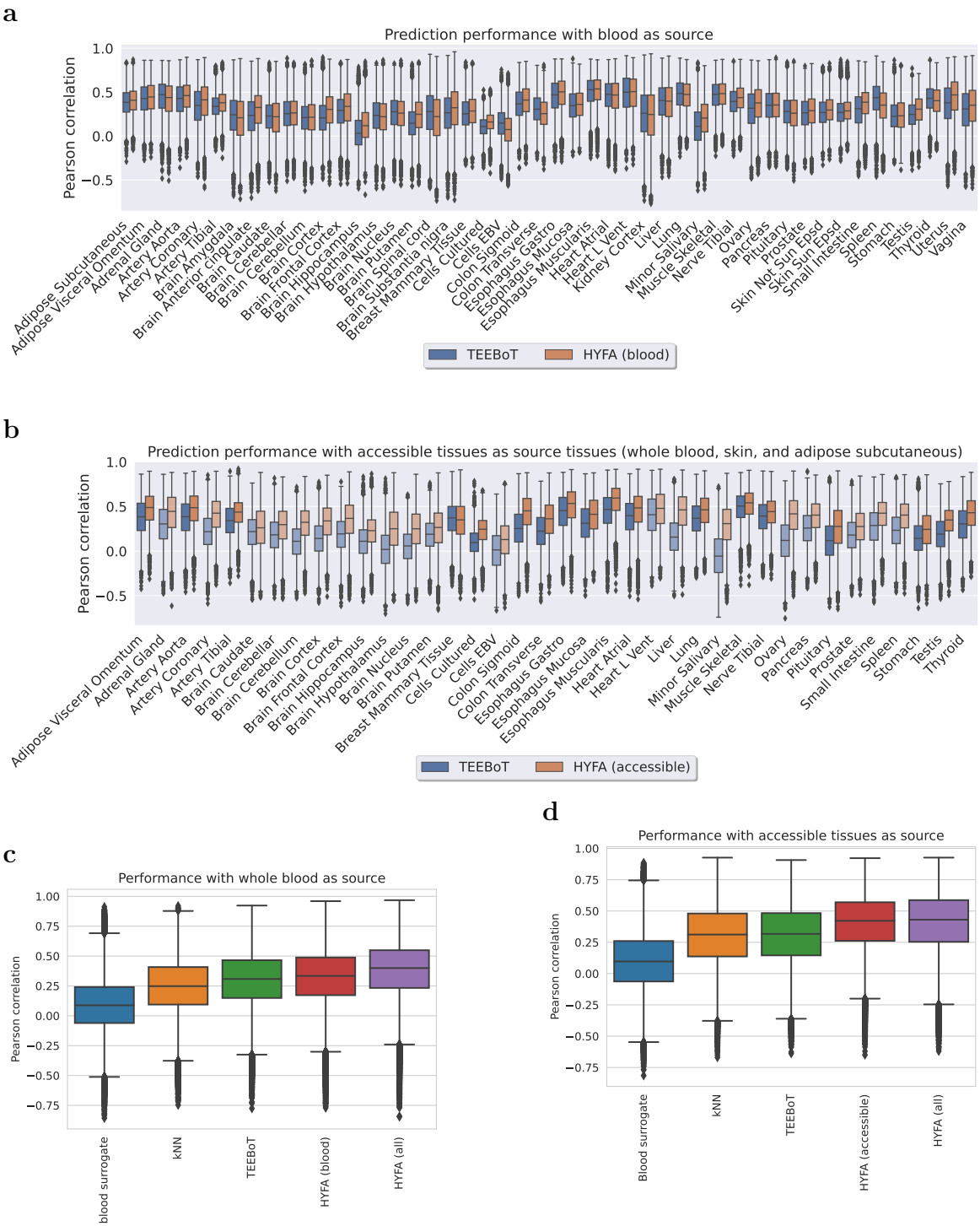


Fig. D.8 Performance comparison with per-gene metrics (*next page*).

Fig. D.8 *(previous page)* Performance comparison across gene expression imputation methods with per-gene metrics (n=12,557 genes; individuals are sampling units). (a, b) Per-tissue comparison between HYFA and TEEBoT when using (a) whole-blood and (b) all accessible tissues (whole blood, skin sun-exposed, skin not sun-exposed, and adipose subcutaneous) as reference. We discarded target tissues represented by less than 25 test individuals. HYFA achieved superior Pearson correlation in (a) 25 out of 48 target tissues when a single tissue was used as reference and (b) all target tissues when multiple reference tissues were considered. For underrepresented target tissues (less than 25 individuals with source and target tissues in the test set), we considered all the validation and test individuals (translucent bars). (c, d) Prediction performance from (c) whole-blood gene expression and (d) accessible tissues as reference. Boxes show quartiles and whiskers depict the distribution range (1.5 times the interquartile range). Mean imputation replaces missing values with per-feature averages. Blood surrogate utilises gene expression in whole blood as a proxy for the target tissue. k-Nearest Neighbours (kNN) imputes missing features with the average of measured values across the k nearest observations (k=20). TEEBoT projects reference gene expression into a low-dimensional space with principal component analysis (PCA; 30 components), followed by linear regression to predict target values. HYFA (all) employs information from all collected tissues. Boxes show quartiles, centerlines correspond to the median, and whiskers depict the distribution range (1.5 times the interquartile range). Outliers outside of the whiskers are shown as distinct points. The top axis indicates the total number of samples for every target tissue.

D.13 Transcription factor enrichment analysis

We applied Gene Set Enrichment Analysis (GSEA) [85] to the gene loadings of HYFA's encoder (Methods). Similar to [257], for a given query gene set, we calculated the maximum running sum of enrichment scores by descending the sorted list of gene loadings for every metagene and factor. We then computed pathway enrichment p-values through a permutation test and employed the Benjamini-Hochberg method to correct for multiple testing. In total, we identified 554 statistically significant enrichments (FDR < 0.05) of TRRUST transcription factors ([258]; Extended Data Figure 6) across all HYFA metagenes (n=50) and factors (n=98).

Among the enriched transcription factors (TFs), we identified important regulators including GATA1 (known to regulate proliferation of immature red blood cells, responsible for delivering oxygen to body tissues [259]), *SPI1* (which controls hematopoietic cell fate; [260]), CEBP TFs (which play an important role in tissue-specific gene expression; [261]), and STAT1, a member of the STAT protein family that drives the expression of many genes [263]. We further observed that the learnt HYFA factors recapitulate synergistic effects among the enriched TFs. For example, GATA1 and SPI1 appear to functionally antagonise each other through physical interaction [307] and were simultaneously enriched in 7 factors (FDR < 0.05; Extended Data Figure 6b). Similarly, IRF1 induces STAT1 activation via phosphorylation [263, 265] and they were enriched together in 10 factors (FDR < 0.05; Extended Data Figure 6b).

D.14 Gene Ontology Biological Process enrichment analysis

We applied Gene Set Enrichment Analysis (GSEA) [85] to the gene loadings of HYFA's encoder (Methods), using gene sets from the Gene Ontology (GO Biological Process; [79]; version of 2021; 6036 gene sets). In total, we identified 9557 statistically significant enrichments (FDR < 0.05) of GO Biological Process terms across all HYFA metagenes (n=50) and factors (n=98), of which 874 corresponded to signaling pathways (Extended Data Figures 7 and 8). Among these, the Type-I Interferon Signaling pathway was enriched the most (GO:0060337; FDR < 0.05 in 308/874 enrichments) followed by Interferon-Gamma-Mediated signaling pathway (GO:0060333; FDR < 0.05 in 202/874 enrichments). Type I interferons (IFNs) are a family of cytokines that bind to a common cell-surface receptor (type I IFN receptor) and activate a variety of signaling cascades. In particular, IFNs are known to turn on STAT (signal transducer and

activator of transcription) complexes, which control the transcription of a large number of target genes [308]. STAT1 (a member of the STAT protein family that plays an important role in regulating the expression of many genes [263]) and IRF1 (a member of the interferon regulatory transcription factor that activates STAT1 among other targets) were highly enriched in our TF enrichment analysis (Extended Data Figure 6).

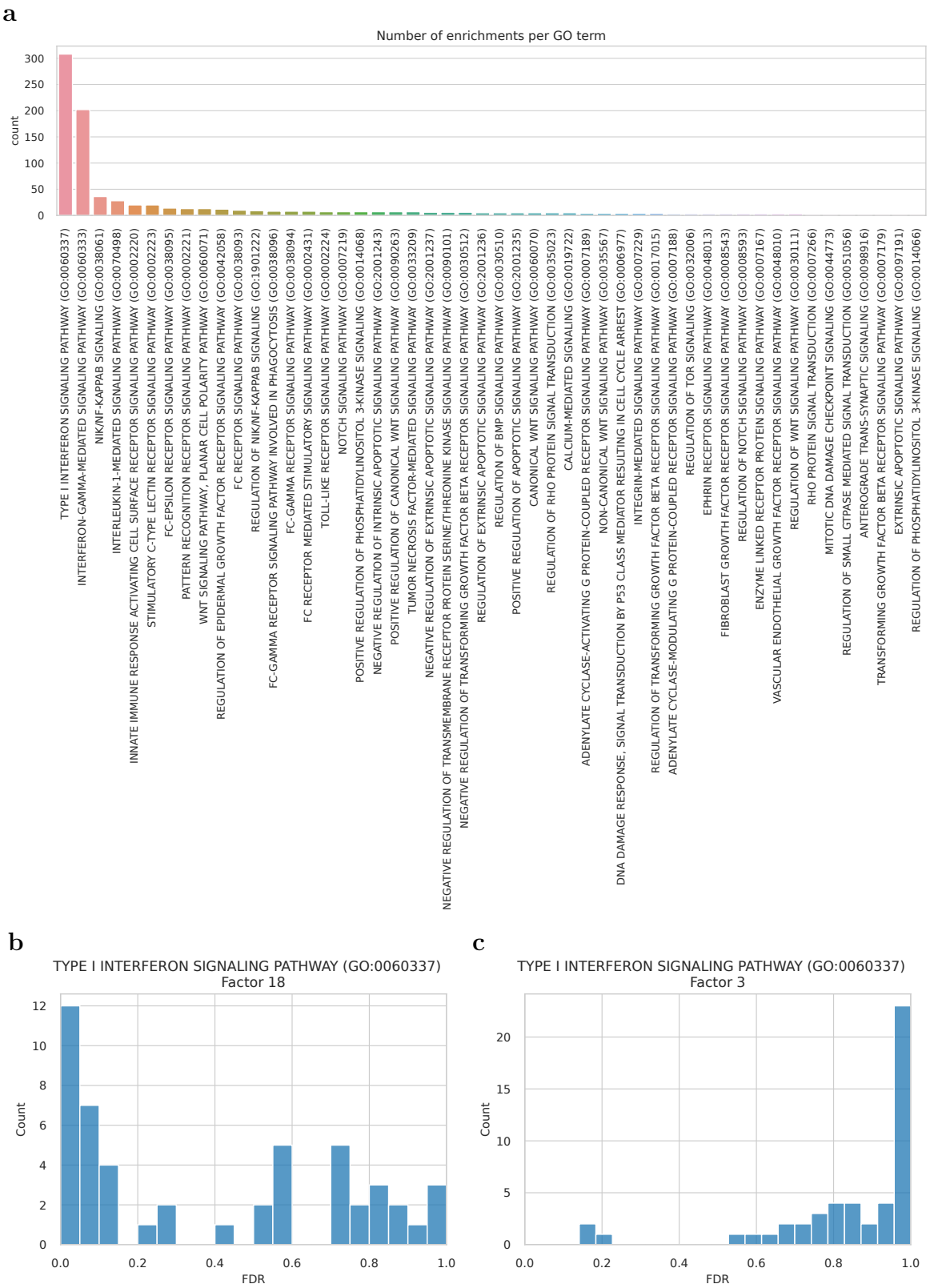


Fig. D.9 GO Biological Process enrichment analysis of metagene-factors (*next page*).

d

Fig. D.9 GO Biological Process enrichment analysis of metagene-factors. For every metagene (n=50) and factor (n=98), we performed Gene Set Enrichment Analysis using the corresponding gene loadings of HYFA's encoder (Methods) and Gene Ontology gene sets (GO Biological Process; [79]; version of 2021) (Enrichr library: *GO_Biological_Process_2021*). (a) Top enriched signaling GO terms, ranked by the total number of metagene-factors in which the terms were enriched (FDR < 0.05). (b, c) FDR distribution of the Type-I Interferon signaling pathway in factor 18 (FDR $<$ 0.05 in 12/50 metagenes) and an arbitrary factor (enriched in 0/50 metagenes). (d) FDR for signaling pathways. For every pathway and factor, we selected the metagene with lowest FDR and depicted statistically significant values (FDR < 0.05). Point sizes are inversely proportional to the FDR values. Type I interferons (IFNs), a family of cytokines that activate a variety of signaling cascades, were the most enriched. We also detected the simultaneous enrichment of interferon IRF1 and STAT1 (a member of the STAT protein family that drives the expression of many target genes [263]) in 10 factors (FDR < 0.05; Extended Data Figure **??**b), consistent with these results.

D.15 HYFA captures differential expression patterns of kidney cancer

We trained HYFA on gene expression data from The Cancer Genome Atlas (TCGA; [309]) processed with the RNAseqDB pipeline [310]. We used HYFA to infer gene expression in kidney tumor sites from the transcriptome measured at the normal tissue adjacent to the tumor (NAT). The NAT tissue is often used as a control in cancer studies, but these regions commonly have phenotypic and morphologic differences with respect to healthy tissue [311]. Genes identified through differential expression analysis on the imputed data overlapped with those detected from the ground truth data (Supplementary Figure D.10). Several of the top differentially expressed genes were predicted with high Pearson correlation (SPAG4: $\rho = 0.631$, BBC3: $\rho = 0.630$, SCARB1: $\rho = 0.593$). Overall, HYFA's imputed gene expression profiles captured differential expression patterns of kidney cancer.

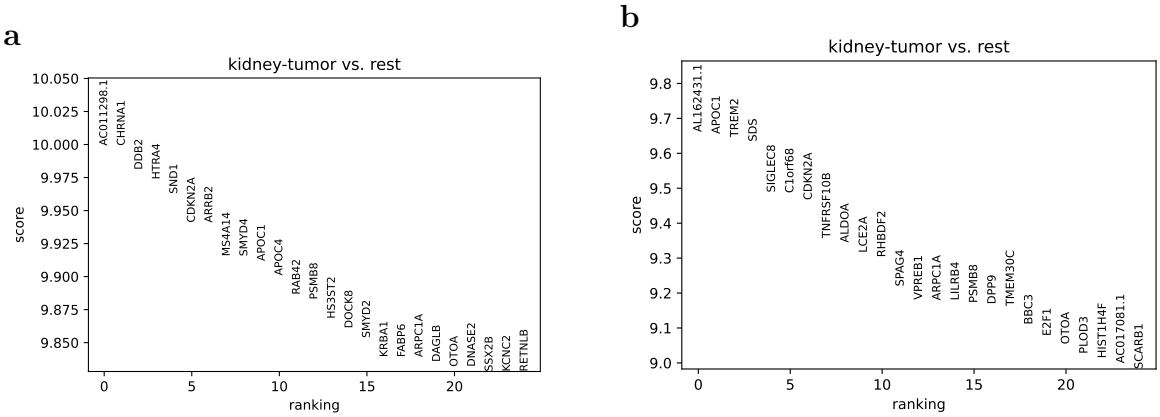


Fig. D.10 HYFA's imputed data captures different expression patterns in kidney cancer (*next page*).

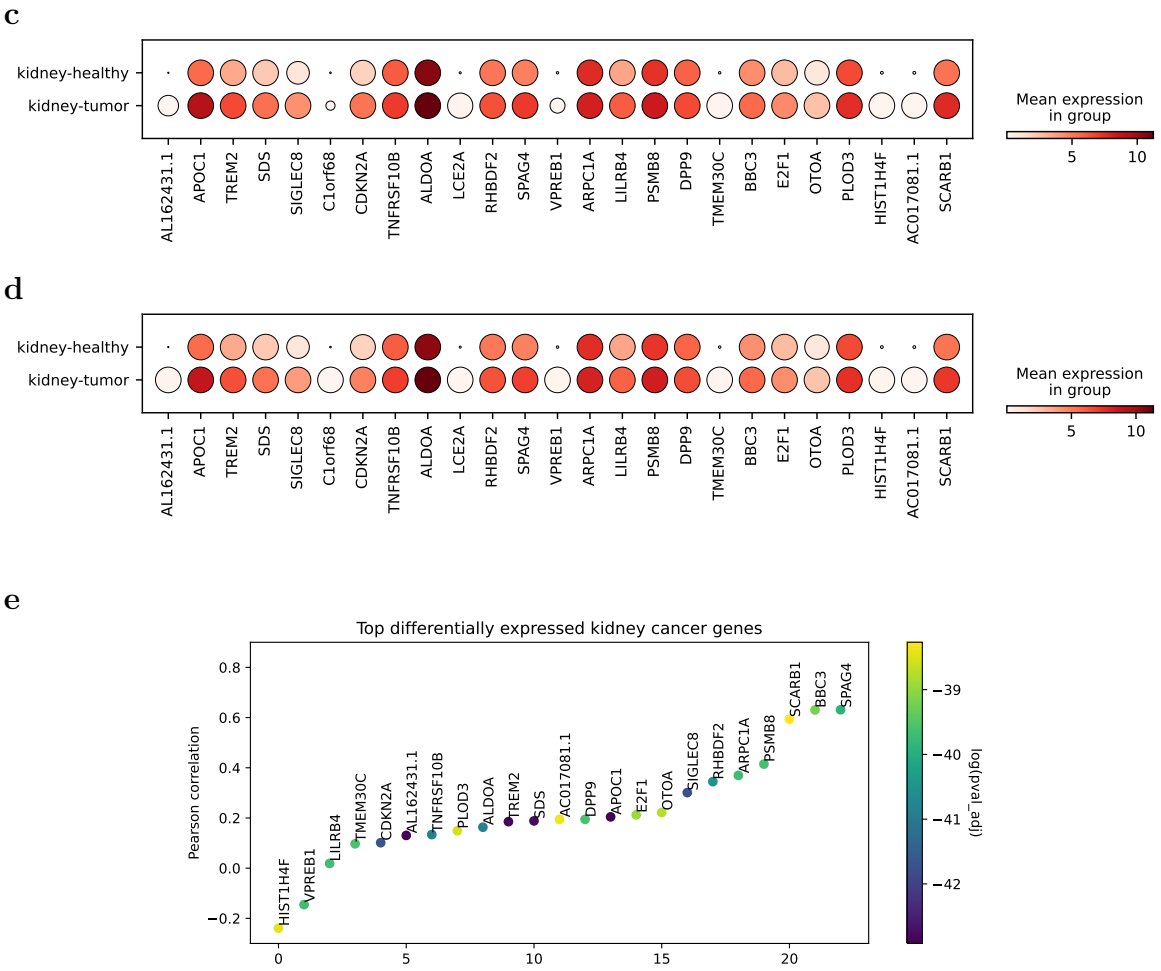


Fig. D.10 HYFA's imputed data captures different expression patterns in kidney cancer. We imputed kidney cancer transcriptome (n=47 test samples) from the gene expression measured at the normal tissue adjacent to the tumor (NAT). We employed a Wilcoxon rank-sum test to rank differentially expressed kidney cancer genes (Scanpy function `scanpy.tl.rank_genes_groups`). We used all the kidney NAT samples as the control group (n=117 control samples). (a, b) Top 25 differentially expressed genes in (a) the imputed data and (b) the real data. (c, d) Average kidney-tumor log-expression profiles of top 25 differentially expressed genes in (c) imputed data and (d) ground truth. The dot sizes are proportional to the number of samples where the gene was expressed. (e) Prediction performance of top 25 differentially expressed genes measured by Pearson correlation. Genes are colored by log-adjusted Benjamini-Hochberg's p-value. Overall, HYFA's imputed profiles captured differential expression patterns of kidney cancer.

D.16 GTEx-v9 train/test splits

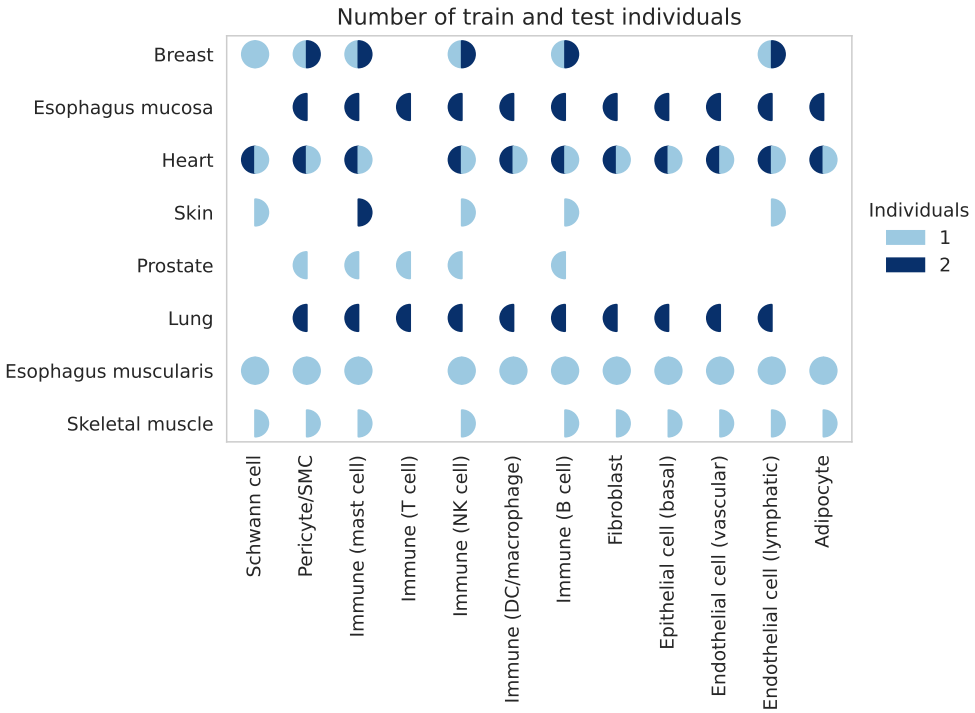


Fig. D.11 Number of train (left semi-circle) and test (right semi-circle) signatures per tissue and cell-type. Each signature corresponds to the aggregated tissue- and cell-type-specific scRNA-seq counts for a given individual. For any combination of tissue and cell-type, there are no more than 2 individual-specific signatures in the same set. Blank semi-circles indicate zero signatures. Note that some signatures (e.g. cell-types in skeletal muscle) are only present in the test set.

D.17 GTEx-v9 predictions with inferred library sizes

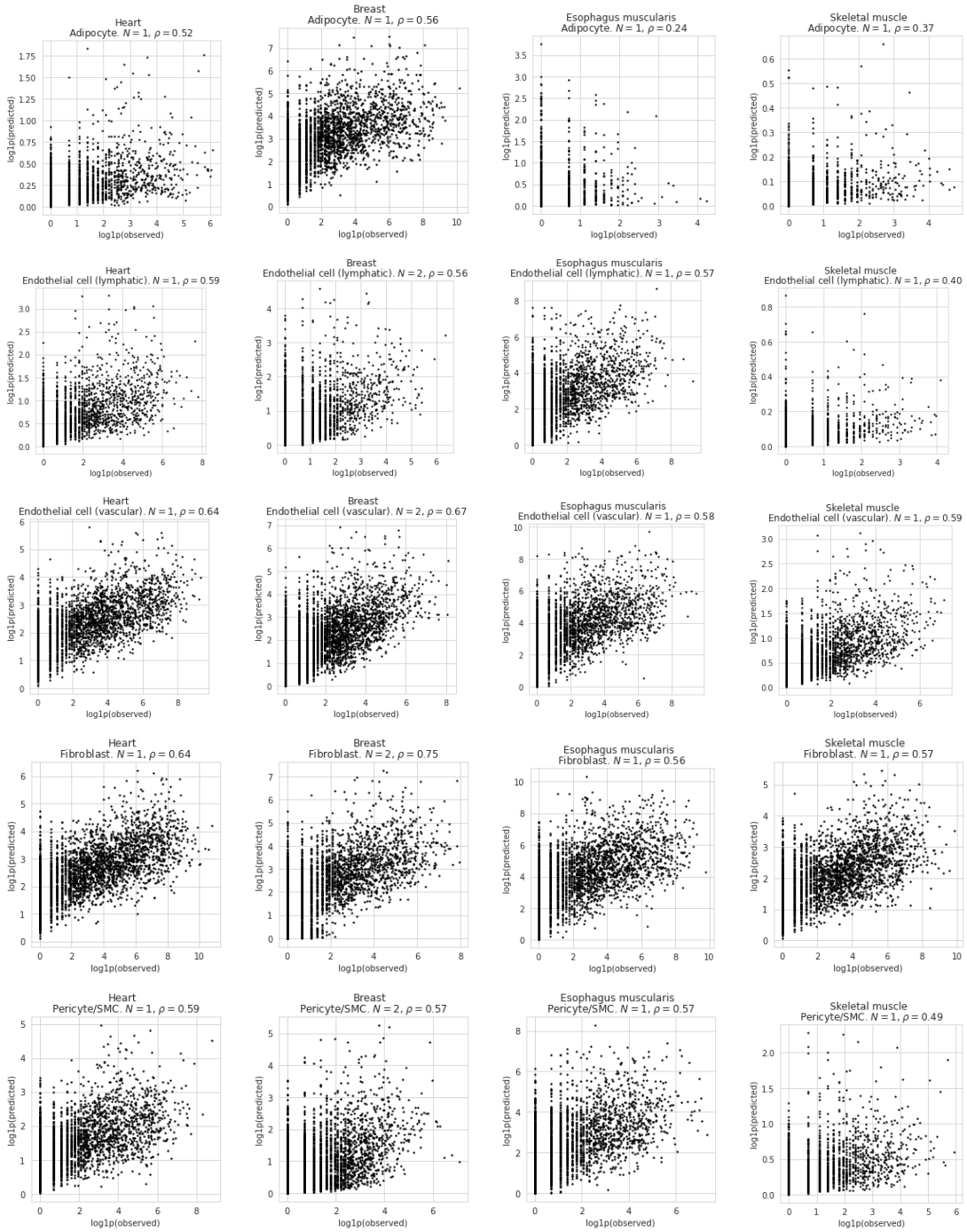


Fig. D.12 Prediction of cell-type signatures. HYFA imputes individual- and tissue-specific cell-type signatures from bulk multi-tissue gene expression. The scatter plots depict the Pearson correlation ρ between the logarithmised ground truth and predicted signatures for N unseen individuals. To predict the signatures, we inferred the library sizes $l_i^{(k,q)}$ and used the observed number of cells $n_i^{(k,q)}$ (Methods).

D.18 Baseline for cell-type signature inference (GTEx-v9)

As a baseline for the cell-type signature inference task, we implemented the following approach:

1. Apply Principal Component Analysis (PCA) to the entire GTEx-v8 bulk transcriptomics dataset ($K = 30$ components), yielding a low-dimensional dataset $\tilde{\boldsymbol{X}}_i \in \mathbb{R}^{|\mathcal{T}(i)| \times K}$ for every individual i.

2. For every target cell-type signature $\boldsymbol{x}_i^{(t,c)}$ of cell-type c, tissue t, and individual i, select the bulk sample $\tilde{\boldsymbol{x}}_i^t$ from $\tilde{\boldsymbol{X}}_i$ matching the individual i and tissue t of the signature.

3. Fit the linear model:

$$\boldsymbol{x}_i^{(t,c)} = \boldsymbol{W}_1 \tilde{\boldsymbol{x}}_i^{(t)} + \boldsymbol{W}_2 \boldsymbol{e}_c + \boldsymbol{b} + \boldsymbol{\epsilon}_i^{(t,c)},$$

   where $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, and $\boldsymbol{b}$ are learnable parameters, $\boldsymbol{e}_c$ is a one-hot vector (1 for cell-type c and 0 otherwise), and $\boldsymbol{\epsilon}_i^{(t,c)}$ is the error term. The signatures $\boldsymbol{x}_i^{(t,c)}$ are normalised by the library size.

4. Predict the unseen cell-type signatures using the learnt model.

Supplementary Table D.4 shows the Pearson correlation between the inferred and ground truth signatures for this baseline as well as the fine-tuned HYFA model (Methods). In summary, we observed that both methods attained comparable results - the baseline achieved a mean Pearson correlation of 0.679 ± 0.012 (mean $\pm$ standard error; baseline) and 0.693 ± 0.021 (mean $\pm$ standard error; fine-tuned HYFA) across signatures. In contrast to this baseline, HYFA is able to utilise information from multiple source tissues. However, HYFA's encoder does not have *a priori* knowledge about the target tissue, potentially leading to information loss. In the future, HYFA's encoder may be extended to extract information specifically relevant for the target tissue and cell types of interest.

Table D.4 Prediction performance on the unseen cell-type signatures measured by Pearson correlation between the log ground truth and log predicted signatures. Baseline corresponds to a method that infers the signatures from the dimensionality-reduced bulk expression measured in the target tissue of the matching individual. For both methods, we used the observed library sizes (i.e. the total counts between the predicted and inferred signatures match).

Tissue	Cell-type	Baseline	HYFA (fine-tuned)
Breast	Adipocyte	0.657	0.749
	Endothelial cell (lymphatic)	0.786	0.840
	Endothelial cell (vascular)	0.774	0.879
	Fibroblast	0.820	0.894
	Immune (DC/macrophage)	0.791	0.834
	Pericyte/SMC	0.767	0.806
Esophagus muscularis	Adipocyte	0.524	0.348
	Endothelial cell (lymphatic)	0.672	0.660
	Endothelial cell (vascular)	0.681	0.683
	Fibroblast	0.667	0.702
	Immune (B cell)	0.634	0.501
	Immune (DC/macrophage)	0.686	0.806
	Immune (NK cell)	0.734	0.714
	Immune (T cell)	0.700	0.716
	Immune (mast cell)	0.632	0.607
	Pericyte/SMC	0.707	0.661
Heart	Adipocyte	0.729	0.711
	Endothelial cell (lymphatic)	0.739	0.821
	Endothelial cell (vascular)	0.707	0.842
	Fibroblast	0.682	0.841
	Immune (B cell)	0.640	0.530
	Immune (DC/macrophage)	0.714	0.836
	Immune (NK cell)	0.765	0.788
	Immune (T cell)	0.751	0.805
	Immune (mast cell)	0.627	0.572
	Pericyte/SMC	0.718	0.782
Skeletal muscle	Adipocyte	0.636	0.549
	Endothelial cell (lymphatic)	0.709	0.711
	Endothelial cell (vascular)	0.734	0.788
	Fibroblast	0.645	0.772
	Immune (DC/macrophage)	0.696	0.770
	Immune (NK cell)	0.684	0.676
	Immune (T cell)	0.685	0.714
	Immune (mast cell)	0.586	0.530
	Pericyte/SMC	0.707	0.680
Skin	Adipocyte	0.581	0.573
	Endothelial cell (vascular)	0.579	0.587
	Fibroblast	0.561	0.518
	Immune (DC/macrophage)	0.488	0.422
	Pericyte/SMC	0.583	0.484

D.19 Cell-type inference in MSK SPECTRUM

We used HYFA to infer cell-type signatures in the MSK SPECTRUM dataset. We downloaded the MSK SPECTRUM data from `https://cellxgene.cziscience.com/collections/4796c91c-9d8f-4692-be43-347b1727f9d8` [312]. We selected the top 3000 highly variable genes using the Scanpy function `sc.pp.highly_variable_genes` and aggregated the single-cell RNA-seq profiles by individual, tissue, and cell-type. After discarding signatures represented by less than 50 cells, we arrived at 1226 individual- tissue- and cell-type-specific signatures from 41 individuals (24 train, 8 validation, 9 test). For a certain individual, we trained HYFA to predict the cell-type signatures of a target tissue as a function of all the available signatures in the remaining tissues. We performed message passing on a 4-uniform hypergraph with individual, tissue, cell-type, and metagene nodes. We optimised the zero-inflated negative binomial likelihood of the target signatures using the observed library size (Chapter 5).

Overall, HYFA attained strong prediction scores (Pearson correlation between log ground truth and log predicted signatures) and captured cell-type-specific gene expression patterns. HYFA-inferred signatures had a strong correlation with the ground truth in most tissues (Supplementary Table D.5) — including transverse colon (average $\rho = 0.91$), intestine (average $\rho = 0.86$) and left ovary (average $\rho = 0.88$) — and cell types — including monocytes (average $\rho = 0.86$), T cells (average $\rho = 0.90$), and plasma cells (average $\rho = 0.80$). Mast cells exhibited comparatively lower correlation (average $\rho = 0.76$). To study whether HYFA captures cell-type specific gene expression patterns, we identified differentially expressed genes from the real signatures using a Wilcoxon rank-sum test (Scanpy function `scanpy.tl.rank_genes_groups`) and then examined the expression of these genes in the inferred signatures (Supplementary Figure D.13). Remarkably, HYFA recovered expression of the main marker genes with high specificity.

Finally, we studied whether the individual- tissue- and cell-type-specific signatures inferred by HYFA can be used to deconvolve *pseudo-bulk* gene expression. For every unseen individual and tissue, we created *pseudo-bulk* samples by aggregating the read counts of all cells in the given tissue. Next, we selected all genes that were both differentially-expressed (Wilcoxon rank-sum adjusted p-value < 0.05) and well-predicted ($R^2 > 0.7$) in the validation set. We then used linear regression (without intercept) to infer the cell-type proportions using (a) the real individual- tissue- cell-type specific signatures, (b) random signatures (i.e. random permutations of the real signatures), and (c) HYFA's inferred signatures. We also considered a uniform baseline (i.e. all cell-types have equal probability). We assessed performance using the mean absolute deviation

Table D.5 Cell-type signature imputation performance in the MSK SPECTRUM dataset, measured by Pearson correlation between the log ground truth and log predicted signatures, with number of individuals in parenthesis.

Tissue	B	T	dendritic	endothelial	epithelial	fibroblast	mast	monocyte	plasma
abdomen	0.84 (5)	0.92 (5)	0.88 (5)	0.90 (5)	0.88 (5)	0.89 (5)	0.71 (5)	0.94 (5)	0.82 (5)
abdominal wall	0.80 (1)	0.91 (1)	0.93 (1)	0.83 (1)	0.85 (1)	0.89 (1)	— (0)	0.90 (1)	0.86 (1)
adnexa of uterus	0.80 (18)	0.91 (18)	0.88 (17)	0.92 (18)	0.88 (18)	0.92 (18)	0.78 (18)	0.94 (18)	0.79 (18)
ascitic fluid	0.83 (25)	0.90 (25)	0.89 (25)	0.47 (6)	0.76 (21)	0.86 (17)	0.52 (13)	0.91 (25)	0.81 (25)
caecum	0.86 (1)	0.91 (1)	0.87 (1)	0.93 (1)	0.88 (1)	0.94 (1)	0.86 (1)	0.95 (1)	0.88 (1)
diaphragm	0.84 (3)	0.92 (3)	0.91 (3)	0.92 (3)	0.90 (3)	0.93 (3)	0.79 (3)	0.94 (3)	0.83 (3)
fallopian tube	0.88 (2)	0.73 (3)	0.90 (2)	0.89 (3)	0.88 (3)	0.89 (3)	0.81 (2)	0.78 (3)	0.78 (3)
intestine	0.86 (9)	0.91 (10)	0.89 (9)	0.90 (11)	0.87 (11)	0.92 (11)	0.73 (9)	0.93 (10)	0.80 (9)
large intestine	0.80 (2)	0.91 (2)	0.67 (2)	0.89 (2)	0.86 (2)	0.91 (2)	0.76 (2)	0.93 (2)	0.74 (2)
left ovary	0.84 (9)	0.92 (9)	0.90 (9)	0.92 (9)	0.88 (9)	0.93 (9)	0.82 (7)	0.95 (9)	0.83 (9)
liver	0.80 (1)	0.90 (1)	0.85 (1)	0.91 (1)	0.90 (1)	0.92 (1)	0.77 (1)	0.94 (1)	0.50 (1)
lymph node	0.88 (2)	0.89 (2)	0.87 (2)	0.88 (2)	0.85 (2)	0.81 (2)	0.68 (2)	0.94 (2)	0.82 (2)
omentum	0.84 (34)	0.90 (34)	0.89 (32)	0.92 (34)	0.87 (34)	0.91 (35)	0.76 (32)	0.93 (34)	0.83 (33)
paracolic gutter	0.87 (1)	0.93 (1)	0.91 (1)	0.94 (1)	0.88 (1)	0.95 (1)	0.84 (1)	0.94 (1)	0.89 (1)
parietal peritoneum	0.87 (1)	0.94 (1)	0.91 (1)	0.94 (1)	0.90 (1)	0.94 (1)	0.89 (1)	0.95 (1)	0.89 (1)
peritoneum	0.84 (14)	0.92 (14)	0.89 (14)	0.87 (12)	0.83 (14)	0.89 (13)	0.79 (11)	0.94 (14)	0.83 (14)
right ovary	0.82 (11)	0.91 (12)	0.88 (10)	0.89 (11)	0.86 (11)	0.86 (12)	0.83 (8)	0.93 (12)	0.79 (10)
transverse colon	0.88 (1)	0.93 (1)	0.92 (1)	0.94 (1)	0.89 (1)	0.93 (1)	0.85 (1)	0.94 (1)	0.90 (1)
urinary bladder	0.36 (1)	0.86 (1)	— (0)	0.88 (1)	0.88 (1)	0.89 (1)	0.60 (1)	0.91 (1)	0.76 (1)

(mAD) between the inferred and ground-truth cell-type proportions (Supplementary Figure D.14). Overall, deconvolution using HYFA's inferred signatures was better than (a) the random signatures and (d) the uniform baselines. In general the per-cell-type absolute deviation scores associated to HYFA's signatures were lower than those of the uniform baseline, with exception to mast cells (mAD= 0.14), consistent with the lower prediction scores for that cell type. Performance using the ground truth signatures was close to perfect. In the future, as single-cell RNA-seq datasets become larger in number of individuals, we expect the resolution of HYFA's inferred signatures to increase, with potential benefits in terms of downstream analysis including deconvolution or cell-type specific eQTL mapping.

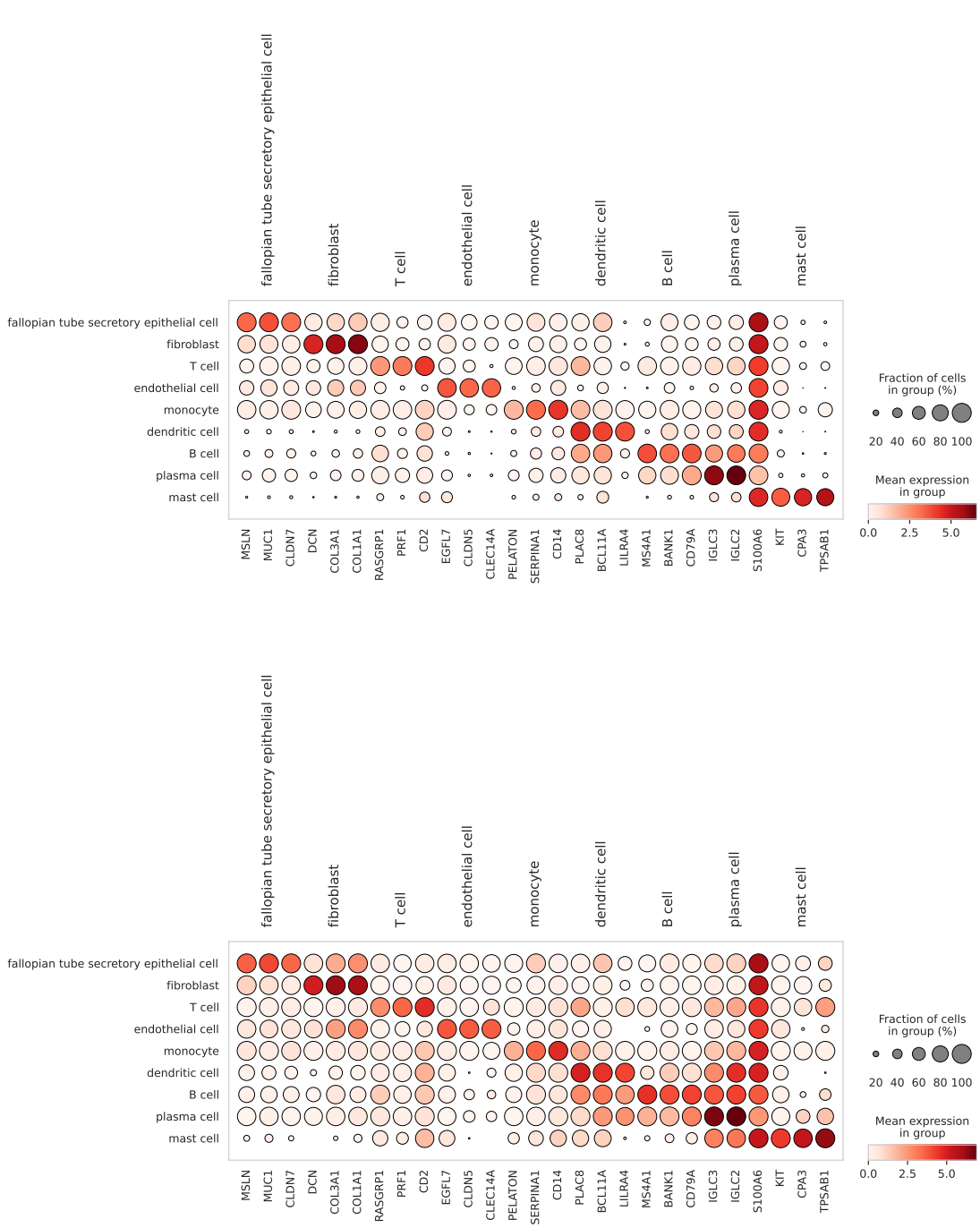


Fig. D.13 Dot plot showing gene expression of top 3 differentially-expressed markers detected from the real signatures. (a) Average gene expression in real signatures. (b) Average gene expression in inferred signatures. Overall, HYFA recovered the main differentially-expressed markers with high specificity.

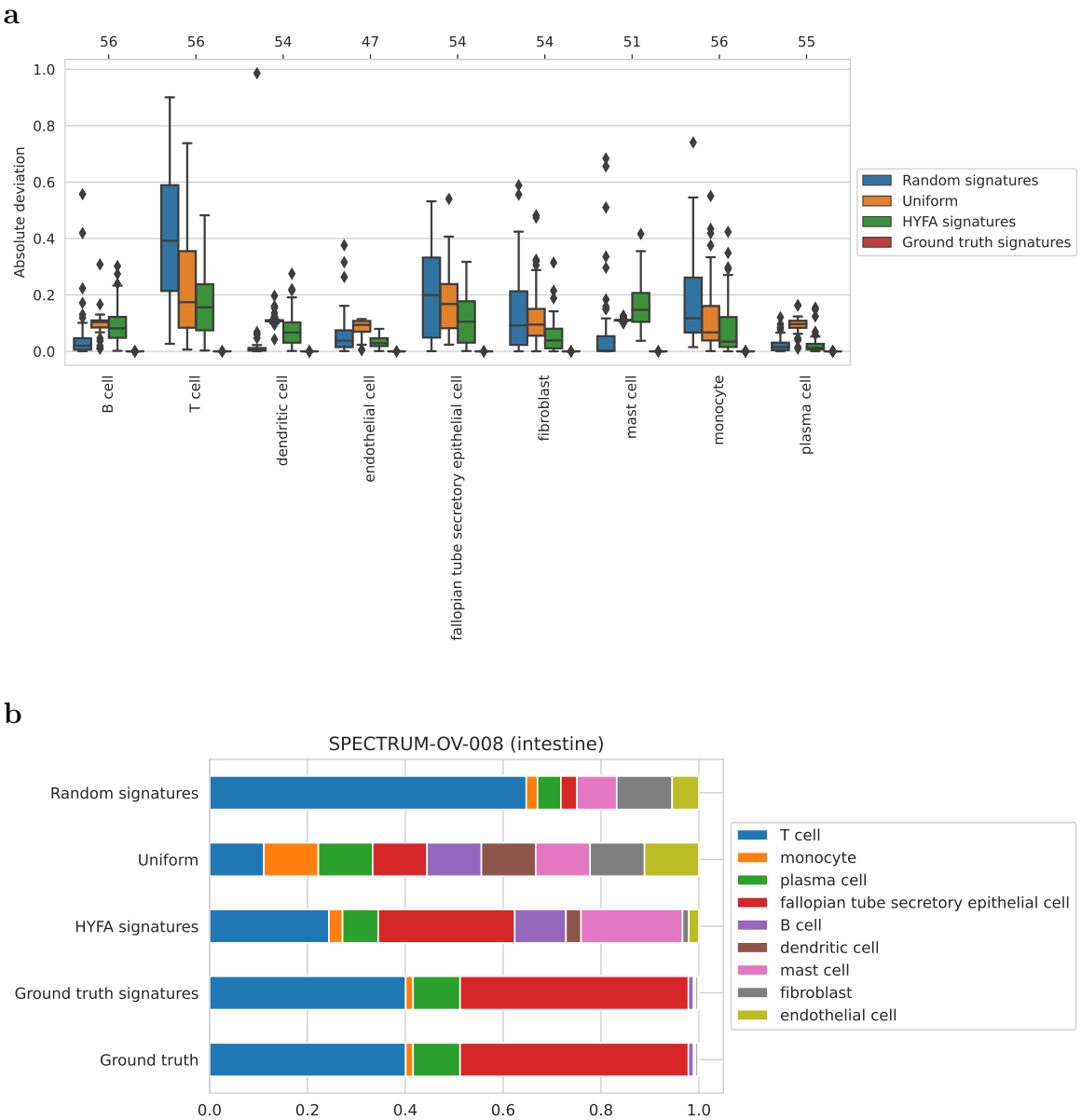


Fig. D.14 Deconvolution performance. We used linear regression to infer cell-type proportions from *pseudo-bulk* gene expression samples using the real individual- tissue-cell-type specific signatures, random signatures (i.e. random permutations of the real signatures), and HYFA's inferred signatures. (a) Absolute deviation between the inferred and ground-truth proportions for every cell-type. Boxes show quartiles, centerlines correspond to the median, and whiskers depict the distribution range (1.5 times the interquartile range). Outliers outside of the whiskers are shown as distinct points. The top axis indicates the total number n of independent samples for every cell type. (b) Deconvolution of a *pseudo-bulk* gene expression sample. For HYFA's signatures, the inferred mast cell fraction is larger than expected, consistent with the fact that prediction performance is lower for this cell type.

Supplementary Information E

Understanding cell-type heterogeneity in tissues from spatial transcriptomics

E.1 Ablation on the number of GNN layers

We studied deconvolution performance of GNN-C2L model variants across different number of layers using the the synthetic dataset from [96] (Tables E.1, E.2, and E.3). We noted that results were generally optimal when using 1-3 layers. The performance dropped for 3+ layers, potentially due the oversmoothing and oversquashing phenomenons [220] which makes it difficult for GNNs to incorporate information from distant neighbours as the aggregation of messages into fixed size vectors, creating an information bottleneck. The optimal number of GNN layers might depend on the tissue architecture, topology of neighbourhood graph, and spot resolution.

Table E.1 Average Pearson R correlation and standard deviation of 5 seeded runs of each model over all spots. Correlation values for subcategories of cell types exhibiting distinct cell abundance patterns are also provided. Bold numbers indicate best performing method for each category of cell types being evaluated. Table credit: Paul Scherer.

Methods	ALL	UHCA	ULCA	RHCA	RLCA
SGC-C2L1	0.699 ± 0.023	0.876 ± 0.008	0.708 ± 0.020	0.883 ± 0.006	0.439 ± 0.041
SGC-C2L2	0.711 ± 0.036	0.890 ± 0.015	0.682 ± 0.027	0.878 ± 0.01	0.458 ± 0.050
SGC-C2L3	0.684 ± 0.063	**0.897 ± 0.019**	0.689 ± 0.030	0.883 ± 0.006	0.421 ± 0.086
SGC-C2L4	0.704 ± 0.025	0.883 ± 0.022	0.673 ± 0.043	0.881 ± 0.009	0.445 ± 0.043
SGC-C2L5	0.701 ± 0.016	0.884 ± 0.015	0.665 ± 0.032	0.882 ± 0.007	0.443 ± 0.034
SGC-C2L6	0.701 ± 0.016	0.884 ± 0.015	0.665 ± 0.032	0.882 ± 0.007	0.443 ± 0.034
GAT-C2L1	**0.737 ± 0.013**	0.885 ± 0.018	0.695 ± 0.032	0.888 ± 0.004	**0.492 ± 0.032**
GAT-C2L2	0.722 ± 0.022	0.879 ± 0.020	0.710 ± 0.042	**0.889 ± 0.004**	0.473 ± 0.029
GAT-C2L3	0.679 ± 0.039	0.872 ± 0.021	**0.723 ± 0.016**	0.887 ± 0.007	0.425 ± 0.052
GAT-C2L4	0.709 ± 0.047	0.878 ± 0.016	0.695 ± 0.024	0.883 ± 0.004	0.474 ± 0.070
GAT-C2L5	0.713 ± 0.050	0.857 ± 0.015	0.698 ± 0.027	0.878 ± 0.009	0.478 ± 0.082
GAT-C2L6	0.715 ± 0.050	0.858 ± 0.016	0.699 ± 0.025	0.878 ± 0.009	0.480 ± 0.082

Table E.2 Average of average Jensen-Shannon divergence (JSD) along with standard deviation of 5 seeded runs of each model. JSD values for subcategories of cell types exhibiting distinct cell abundance patterns are also provided. Bold numbers indicate best performing method for each category of cell types being evaluated. Table credit: Paul Scherer.

Methods	ALL	UHCA	ULCA	RHCA	RLCA
SGC-C2L1	0.446 ± 0.006	0.224 ± 0.011	0.460 ± 0.007	0.368 ± 0.005	0.493 ± 0.009
SGC-C2L2	0.443 ± 0.007	0.208 ± 0.021	0.467 ± 0.010	0.371 ± 0.009	0.489 ± 0.007
SGC-C2L3	0.447 ± 0.011	**0.199 ± 0.017**	0.463 ± 0.006	0.369 ± 0.007	0.499 ± 0.015
SGC-C2L4	0.448 ± 0.006	0.216 ± 0.019	0.472 ± 0.014	0.375 ± 0.008	0.494 ± 0.009
SGC-C2L5	0.448 ± 0.005	0.207 ± 0.022	0.473 ± 0.010	0.375 ± 0.007	0.493 ± 0.008
SGC-C2L6	0.448 ± 0.005	0.207 ± 0.022	0.473 ± 0.010	0.375 ± 0.007	0.493 ± 0.008
GAT-C2L1	**0.435 ± 0.003**	0.209 ± 0.021	0.458 ± 0.014	0.369 ± 0.001	**0.482 ± 0.006**
GAT-C2L2	0.438 ± 0.006	0.223 ± 0.017	0.458 ± 0.014	0.363 ± 0.002	0.486 ± 0.005
GAT-C2L3	0.447 ± 0.008	0.222 ± 0.025	0.450 ± 0.009	**0.356 ± 0.004**	0.496 ± 0.011
GAT-C2L4	0.441 ± 0.010	0.215 ± 0.018	0.452 ± 0.010	0.358 ± 0.004	0.487 ± 0.013
GAT-C2L5	0.445 ± 0.015	0.243 ± 0.017	**0.448 ± 0.009**	0.362 ± 0.007	0.492 ± 0.020
GAT-C2L6	0.444 ± 0.015	0.242 ± 0.017	0.448 ± 0.009	0.362 ± 0.007	0.491 ± 0.020

Table E.3 Average AUPRC scores and standard deviation of 5 seeded runs of each model over all spots. Scores for subcategories of cell types exhibiting distinct cell abundance patterns are also provided. Bold numbers indicate best performing method for each category of cell types being evaluated. Table credit: Paul Scherer.

Methods	ALL	UHCA	ULCA	RHCA	RLCA
SGC-C2L1	0.719 ± 0.002	0.977 ± 0.004	0.646 ± 0.006	0.861 ± 0.001	0.719 ± 0.002
SGC-C2L2	0.716 ± 0.003	0.978 ± 0.001	0.644 ± 0.006	0.860 ± 0.001	0.716 ± 0.003
SGC-C2L3	0.710 ± 0.002	**0.979 ± 0.002**	0.649 ± 0.005	0.852 ± 0.001	0.710 ± 0.002
SGC-C2L4	0.701 ± 0.004	0.972 ± 0.003	0.639 ± 0.007	0.845 ± 0.005	0.701 ± 0.004
SGC-C2L5	0.701 ± 0.007	0.975 ± 0.003	0.633 ± 0.009	0.848 ± 0.005	0.701 ± 0.007
SGC-C2L6	0.701 ± 0.007	0.975 ± 0.003	0.633 ± 0.009	0.848 ± 0.005	0.701 ± 0.007
GAT-C2L1	0.722 ± 0.002	0.978 ± 0.004	0.664 ± 0.004	0.858 ± 0.003	0.722 ± 0.002
GAT-C2L2	**0.726 ± 0.001**	0.977 ± 0.003	0.665 ± 0.007	0.865 ± 0.001	**0.726 ± 0.001**
GAT-C2L3	0.721 ± 0.003	0.970 ± 0.003	**0.679 ± 0.006**	**0.870 ± 0.002**	0.721 ± 0.003
GAT-C2L4	0.710 ± 0.003	0.968 ± 0.002	0.670 ± 0.006	0.867 ± 0.001	0.710 ± 0.003
GAT-C2L5	0.700 ± 0.002	0.959 ± 0.001	0.652 ± 0.010	0.865 ± 0.001	0.700 ± 0.002
GAT-C2L6	0.702 ± 0.003	0.961 ± 0.003	0.652 ± 0.009	0.865 ± 0.001	0.702 ± 0.003