Machine Learning Methods for High Throughput Biological Data

by

Michael A. Murphy

B.A.Sc. Engineering Science, University of Toronto, 2017

Submitted to the Computational and Systems Biology PhD Program
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN COMPUTATIONAL AND SYSTEMS BIOLOGY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2024

Authored by .....
Michael A. Murphy
Computational and Systems Biology PhD Program
November 30, 2023

Certified by .....
Ernest Fraenkel
Professor of Biological Engineering, Thesis Supervisor

Certified by .....
Stefanie S. Jegelka
Associate Professor of Computer Science, Thesis Supervisor

Accepted by .....
Ernest Fraenkel
Professor of Biological Engineering
Co-director, Computational and Systems Biology Graduate Program

Machine Learning Methods for High Throughput Biological Data

by

Michael A. Murphy

Submitted to the Computational and Systems Biology PhD Program
on November 30, 2023 in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN COMPUTATIONAL AND SYSTEMS BIOLOGY

ABSTRACT

Machine learning is becoming a pivotal tool in the analysis of datasets generated from high-throughput biological omics experiments. However, omics data introduces distinctive algorithmic challenges that set it apart from other domains where machine learning is applied. These challenges encompass issues such as limited data availability, complex noise, ambiguities in representation, and the absence of definitive ground truth for validation. In this thesis, I present three examples of machine learning applications to different omics modalities in which I address these challenges. In my first project, I develop an approach for contrastive representation learning with immunohistochemistry images, which suffer complex technical and biological noise that render generic approaches ineffective; and I demonstrate how this approach can be combined with noisy labels derived from transcriptomics to derive an effective classifier of cell-type specificity. In my second project, I consider the problem of predicting mass spectra of small molecules: previous methods suffer from a tradeoff between capturing high-resolution mass information and a tractable learning problem, which I resolve by introducing a novel representation of the output space. In my third project, I perform gene regulatory network inference using a number of different single-cell sequencing platforms, and carry out a quantitative comparison of these technologies. In summary, this thesis showcases the difficulties that arise in applying modern machine learning approaches to high-throughput biological measurements, and empirical case studies of how these difficulties may be overcome.

Thesis supervisor: Ernest M. Fraenkel
Title: Professor of Biological Engineering

Thesis supervisor: Stefanie S. Jegelka
Title: Associate Professor of Computer Science

Acknowledgments

I wish to express my gratitude to the advisors, mentors, colleagues, friends and family that have been instrumental in my journey thus far.

I first thank my co-advisors: Prof. Ernest Fraenkel, for his scholarly and professional guidance, patience, and accommodation as I navigated the initially very foreign world of biological research; and Prof. Stefanie Jegelka, for her intellectual curiosity, rigour, and charity in helping me to explore problems often far removed from ML theory, and especially for the many hours spent whiteboarding problem formulations and derivations, which gave me the confidence to engage in machine learning research with world-class scientists.

My gratitude also extends to my committee members, Profs. Caroline Uhler and Douglas Lauffenburger, for their valuable input throughout the development of this thesis; and to Prof. Nikolai Slavov, for generously providing his time and feedback as an external member.

I also thank my current and former peers in both of my labs: the close friendship and collegiality of those in the Fraenkel group has made my PhD journey both enjoyable and rewarding. In the Jegelka group, I have been privileged to engage with some of the brightest minds in machine learning and have benefited immensely from it. The relationships I have formed with my colleagues here at MIT have made this chapter of my life difficult to close.

Outside of MIT, I would like to acknowledge mentors who have had a profound impact on my journey: notably, Drs. Tom Butler, David Healey, Tobias Kind, and the team at Enveda Biosciences, whose collaboration also facilitated a major part of this thesis. I credit my former colleagues in the software research group at AB Sciex with first introducing me to computational biology and helping me to recognize my potential in applied mathematical research. Other important collaborators and mentors whom I wish to thank, some of whose work appears in this thesis and others whose work inspired it, include: Drs. Joshua Levin and Sami Farhi of the Broad Institute, and the other members of the STWG; Dr. Kevin Yang of Microsoft Research; Profs. Connor Coley, David Sontag, Eric Alm, and Alex Shalek of MIT; Prof. Shawn Davidson of Northwestern; and Profs. Mel Feany and Clemens Scherzer of Harvard Medical School.

I am also deeply indebted to my undergraduate research supervisors at the University of Toronto, Dr. Robert Healey and Prof. Greg Evans, who laid the foundation for my career and allowed me a very early introduction to machine learning in the natural sciences.

Beyond my professional life, I am very grateful for the unwavering support of friends from graduate school, college, and my high school days; as well as the love and encouragement from my siblings, Caroline and Daniel.

Lastly, my deepest gratitude goes to my parents, John and Sandra. Their hard work and sacrifice have allowed me to be where I am today, and none of this would have been possible without them.

Contents

Chapter 1

Introduction

1.1 Omics technology

Evolution has given rise to some of the most complex systems known to science. Our efforts to understand the molecular mechanisms through which living organisms function – and importantly their dysfunction in disease – have been greatly aided by the development of 'omics' technologies: high-throughput biological experiments that detect and quantify large numbers of molecular species. These permit unbiased testing of biological hypotheses at scale, and offer the potential to detect multivariate, nonlinear relationships reflecting macroscopic biological states and microscopic biophysical interactions.

Omics technologies are broadly classified by the classes of biomolecules that they measure: including but not limited to genomics [1], epigenomics [2], transcriptomics [3], proteomics [4], and metabolomics [5]. Each of these approaches captures a certain step in the classical picture – the 'central dogma' – of molecular biology (Figure 1.1).

Particular omics technologies vary widely in their implementations – ranging from single-cell sequencing to fluorescence imaging to mass spectrometry – and can require incompatible sample preparation protocols that render joint measurements impractical. While it is therefore easiest to measure each of these molecular layers in isolation from one another, the evolutionary processes that give rise to biological systems do not abide by such strict compartmentalization. Rather, it is well known that each of these layers interact with the others through dense feedback loops that span multiple spatial and temporal scales [7].

A fuller characterization of a biological system therefore necessitates measurement and integration of signals from multiple omics modalities. The design and analysis of such experiments is difficult and remains far from solved, and often necessitates approaches and expertise orthogonal to those within any particular omics modality [6]. The term 'multi-omics' encompasses the various research efforts involved in both acquiring such data to study a particular biological system, and the development of technologies and algorithms to generate and analyze it.

1.2 Challenges in machine learning for omics

Omics technologies produce vast quantities of data. Machine learning presents a natural analytical tool for this setting, and is by now routinely applied to a diverse array of computational problems in biology [8, 9]. However, when a new biological research question arises, this application is rarely straightforward: major challenges arise in both formulating hypotheses of biological interest as machine learning problems, and conversely in adapting state-of-the-art approaches from the machine learning literature to the biological context. Here, we highlight four broad themes that capture a number of challenges particular to and commonly encountered by machine learning investigations on high-throughput biological data. (Elaboration on specific tasks is deferred to their respective chapters.)

Limited data

Biological data is costly to generate. While large-scale efforts in natural language processing and computer vision can rely on readily-available corpora of text [10] and images [11] scraped from the Internet, and may be annotated usefully by humans without advanced domain expertise, biological datasets are generated via experimentation. Development and execution of a biological experiment is a complex process that can take months to years, and requires substantial time commitment from highly-trained experts, funding, and ethics approval. The data generated by a biological experiment cannot generally be considered 'labelled' either: even when well-defined, ground-truth labels in biology are rarely available, often subjective, and can demand even further confirmatory experimentation [12]. Practically, this not only makes machine learning problems harder: the frequent absence of standard benchmark datasets, against which different algorithms can be objectively compared, also makes the process of machine learning research itself substantially more difficult [13, 8, 14].

Limitations in sample throughput of biological assays present a major challenge for machine learning. The particular molecular layer being measured is a major determinant: while thousands of single-cell transcriptomes and epigenomes can by now be measured commercially [15], single-cell proteomics has only recently begun development [16], and broad sampling of the metabolome remains only feasible at the level of bulk samples [17]. There is also often a tradeoff between measurement throughput and fidelity: sequencing the full length of an mRNA transcript, instead of solely the 3'- or 5'-end, comes at an order of magnitude cost in cell throughput [18]; multiplexed immunofluorescence imaging can measure protein localization at subcellular resolution in whole tissues [19], but can only do so for tens of proteins at a time; and the number of reads to acquire per cell is a major design consideration in sequencing experiments [3]. Additionally, whether an experiment achieves low or high sample throughput is context-dependent: the sample size is determined by what constitute the observational units of interest, be those individual cells, distinct perturbations, or unique patients [20].

Living organisms are complex nonlinear dynamical systems. Unfortunately, most high-throughput omics measurements are destructive, meaning we cannot directly observe high-dimensional cellular trajectories and must rely on reconstructions [21]. Even when time-resolved measurement is a possibility, the sampling frequency at which we can measure molecular variables is not generally the timescale at which the system evolves [22], nor

can we generally observe all relevant variables simultaneously [6]. Given that no single omics technology can capture every biomolecule, critical variables remain latent; this poses challenges for learning and interpretation, especially when correlations between measured variables result from latent confounders or mediators [23].

Noise

Omics measurements exhibit complex noise at multiple scales. This noise has both biological and technical origins: while the former is often of scientific interest, as it serves functional roles [24] and facilitates system identification [25], the latter is usually considered a nuisance. Unfortunately, it is often difficult to distinguish one from the other: as concentrations of biomolecules in the cell span several orders of magnitude [26], noise fluctuations in a particularly abundant molecular species can exceed systematic shifts in the abundance of a rarer one between two conditions of interest. The chemical and physical processes that biological samples undergo in preparation for omics analysis can substantially perturb their state [18, 27], and variability in these processes can manifest across experiments as batch effects [28]. The magnitude of these technical effects can sometimes greatly exceed biological effects between conditions of interest [29], and can affect generalization of machine learning models [30]; moreover, there is a risk of learning spurious correlations when biological and technical effects are confounded [31].

Sparsity poses a related challenge. Many omics measurements, notably single-cell sequencing, operate via counting molecules. As most molecular species are rare, there are fundamental difficulties in sampling deeply enough to ensure a particular signal-to-noise for all variables under observation [6]. In practice, this means single-cell sequencing data is usually very sparse. Whether this sparsity is explainable solely through biological noise, or arises from zero inflation processes during measurement, is technology-dependent and has been the subject of considerable debate [32, 33]. Care must therefore be taken to account for the origins of zeros when designing loss functions [34].

Label noise also poses difficulties. Some of the largest available omics datasets are composed of submissions from the scientific community, and do not undergo extensive curation to ensure consistency across studies [35]. Observations labelled as nominally representing the same molecular variables can in fact describe different molecular species, causing obvious problems for generalization and making meta-analyses difficult [17]; and important covariates necessary to fully specify an experimental condition or measurement process are often missing entirely [36]. Additionally, as there are classes of molecules and interactions that any given measurement technology will be unable to observe, care must taken when generating labels (e.g. from orthogonal omics or low-throughput experimentation) to decouple measurement limitations from the estimated performance of a machine learning algorithm.

Finally, direct measurement of the molecular variables involved in a biological process of interest often proves challenging. As a result, we often resort to measuring an imprecise proxy, such as the mRNA encoding a transcription factor, instead of the nuclear abundance of the protein [37]; or a fluorescence signal from an antibody complementary to a protein of interest, with potential off-target binding [19].

Representation

Omics datasets span a wide range of modalities: beyond observations-by-features matrices, omics experiments can yield sequences, graphs, images, time series; more exotic data modalities such as mass spectra, contact maps, hyperspectral images; and combinations of the former. Not all of these formats are natively conducive to machine learning: transforming them into 'friendlier' mathematical objects can require complex signal processing pipelines that introduce difficult machine learning problems in their own right.

Certain omics modalities can be equivalently represented in multiple spaces: for example, a mass spectrum can be represented as a list of real-valued tuples, a fixed-length vector, a probability distribution over chemical formulas, or a weighted directed acyclic graph [36]. The optimal choice of input and output representation (prior to the learning process itself) for a given problem is therefore often not obvious, and necessitates knowledge of the measurement technology and biological domain. This extends beyond familiar concerns of feature engineering and selection in machine learning, and (as we demonstrate in Chapter 3 of this thesis) can greatly influence predictive accuracy and computational efficiency.

Representation learning also encounters unique challenges with biological data that arise from the preceding discussion on noise. Dimensionality reduction is an important tool for exploratory analysis of high-dimensional omics datasets [38, 39], but application of generic approaches will reflect whatever the major factors of variability are in the data – which are not always biological in origin [29]. As we demonstrate in Chapter 1 of this thesis, robustness of learned representations to technical noise is important for both effective visualization and performance on downstream tasks, but can be nontrivial to achieve when noise is complex.

Implementation

Substantial practical difficulties can arise when implementing and training biological machine learning models, which often demand the researcher be an effective software engineer in addition to their simultaneous roles of machine learning expert and biologist.

Omics datasets can be very large, especially in genomics [40] and imaging [41]. As with other applied machine learning domains, developing deep learning models on terabyte-scale datasets requires access to and fluency with distributed training on high-performance computing clusters. While packages such as PyTorch Lightning [42] facilitate distributed training, data storage and streaming during training is much less 'plug-and-play' and can present unexpected bottlenecks to researchers inexperienced in software engineering and parallel programming.

An orthogonal issue is the availability and quality of libraries for parsing, storing, and processing data, which vary widely across different omics. While single-cell transcriptomics research benefits greatly from mature open-source packages such as ScanPy [43], it is something of an exception: datasets produced by other omics – for example mass spectrometry, where data formats are non-standardized and vendor-specific [35] – frequently necessitate ad-hoc solutions, for which deep understanding of the measurement technology is essential. This difficulty is compounded when working with multi-omics data, which demands familiarity with various software packages (ranging from commercial software packages to academic research code), understanding of the domain-specific concepts therein, and fluency

in debugging across multiple programming languages [44].

1.3 Overview of thesis

This thesis presents three chronologically-ordered case studies in developing machine learning algorithms for high-throughput biological data modalities.

In Chapter 2, I develop an approach for self-supervised representation learning from large-scale immunohistochemistry datasets. This is an imaging omics modality that exhibits complex technical and biological noise, rendering state-of-the-art computer vision approaches ineffective. I introduce modifications to the sampling schemes used in contrastive learning that exploit the structure of biological experiments, and demonstrate this yields representations that capture biological semantics and are robust to technical confounding. I then demonstrate how these representations can be combined with independent single-cell transcriptomics data to derive an effective image classifier of cell-type specificity.

In Chapter 3, I present a novel model for predicting tandem mass spectra of small molecules, which presents an important computational bottleneck in metabolomics. Previous approaches suffer from a tradeoff between capturing high-resolution mass information and posing a tractable learning problem. I resolve this tradeoff by introducing a novel representation of the output space that exploits latent discrete structure of chemical formulas. I then discover an effective constant-sized empirical approximation, which permits applying an efficient graph neural network architecture to the problem. This yields state-of-the-art performance in both spectrum prediction, and in using these predicted spectra to carry out compound retrieval of unknown spectra against a database of known structures.

In Chapter 4, I study the problem of multi-omics experimental design, specifically through the lens of statistically inferring gene regulatory networks. In that work, carried out as part of the Standards and Technologies Working Group of the Human Cell Atlas, I carry out a controlled comparison of regulatory networks inferred from four different single-cell sequencing assays. As no ground truth is available for this task, I develop quantitative evaluation criteria that combine biological priors about gene regulation with orthogonal measurements gathered from external datasets. I observe different predictive performance between models trained on single-nucleus versus single-cell expression, which I connect to differences in nuclear and cytoplasmic mRNA localization; and a tradeoff between predictive performance and biological plausibility of inferred regulatory edges, suggesting influence of factors beyond TF-target interactions.

In summary, each section of the thesis showcases difficulties that arise in practical application of modern machine learning approaches to high-throughput biological measurements generated in pursuit of real-world research questions, and specific examples of how these challenges may be overcome.

References

[1] Charles Gawad, Winston Koh, and Stephen R. Quake. “Single-cell genome sequencing: current state of the science”. In: *Nature Reviews Genetics* 17.3 (Jan. 2016), pp. 175–

188. ISSN: 1471-0064. DOI: 10.1038/nrg.2015.16. URL: http://dx.doi.org/10.1038/nrg.2015.16.

[2] Stephen J. Clark et al. "Single-cell epigenomics: powerful new methods for understanding gene regulation and cell identity". In: *Genome Biology* 17.1 (Apr. 2016). ISSN: 1474-760X. DOI: 10.1186/s13059-016-0944-x. URL: http://dx.doi.org/10.1186/s13059-016-0944-x.

[3] Ashraful Haque et al. "A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications". In: *Genome Medicine* 9.1 (Aug. 2017). DOI: 10.1186/s13073-017-0467-4. URL: https://doi.org/10.1186/s13073-017-0467-4.

[4] Hayley M. Bennett et al. "Single-cell proteomics enabled by next-generation sequencing or mass spectrometry". In: *Nature Methods* 20.3 (Mar. 2023), pp. 363–374. ISSN: 1548-7105. DOI: 10.1038/s41592-023-01791-5. URL: http://dx.doi.org/10.1038/s41592-023-01791-5.

[5] Caroline H. Johnson, Julijana Ivanisevic, and Gary Siuzdak. "Metabolomics: beyond biomarkers and towards mechanisms". In: *Nature Reviews Molecular Cell Biology* 17.7 (Mar. 2016), pp. 451–459. ISSN: 1471-0080. DOI: 10.1038/nrm.2016.25. URL: http://dx.doi.org/10.1038/nrm.2016.25.

[6] Sonia Tarazona, Angeles Arzalluz-Luque, and Ana Conesa. "Undisclosed, unmet and neglected challenges in multi-omics studies". In: *Nature Computational Science* 1.6 (June 2021), pp. 395–402. ISSN: 2662-8457. DOI: 10.1038/s43588-021-00086-z. URL: http://dx.doi.org/10.1038/s43588-021-00086-z.

[7] George Orphanides and Danny Reinberg. "A Unified Theory of Gene Expression". In: *Cell* 108.4 (Feb. 2002), pp. 439–451. ISSN: 0092-8674. DOI: 10.1016/s0092-8674(02)00655-4. URL: http://dx.doi.org/10.1016/s0092-8674(02)00655-4.

[8] Joe G. Greener et al. "A guide to machine learning for biologists". In: *Nature Reviews Molecular Cell Biology* 23.1 (Sept. 2021), pp. 40–55. ISSN: 1471-0080. DOI: 10.1038/s41580-021-00407-0. URL: http://dx.doi.org/10.1038/s41580-021-00407-0.

[9] Nicolae Sapoval et al. "Current progress and open challenges for applying deep learning across the biosciences". In: *Nature Communications* 13.1 (Apr. 2022). ISSN: 2041-1723. DOI: 10.1038/s41467-022-29268-7. URL: http://dx.doi.org/10.1038/s41467-022-29268-7.

[10] Hugo Touvron et al. "LLaMA: Open and Efficient Foundation Language Models". In: *CoRR* abs/2302.13971 (2023). DOI: 10.48550/arXiv.2302.13971. arXiv: 2302.13971. URL: https://doi.org/10.48550/arXiv.2302.13971.

[11] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *International Journal of Computer Vision (IJCV)* 115.3 (2015), pp. 211–252. DOI: 10.1007/s11263-015-0816-y.

[12] Aditya Pratapa et al. "Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data". In: *Nature Methods* 17.2 (Jan. 2020), pp. 147–154. DOI: 10.1038/s41592-019-0690-6. URL: https://doi.org/10.1038/s41592-019-0690-6.

[13] Christopher Lance et al. "Multimodal single cell data integration challenge: Results and lessons learned". In: *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*. Ed. by Douwe Kiela, Marco Ciccone, and Barbara Caputo. Vol. 176. Proceedings of Machine Learning Research. PMLR, Dec. 2022, pp. 162–176. URL: https://proceedings.mlr.press/v176/lance22a.html.

[14] Daniel Machado and Markus Herrgård. "Systematic Evaluation of Methods for Integration of Transcriptomic Data into Constraint-Based Models of Metabolism". In: *PLoS Computational Biology* 10.4 (Apr. 2014). Ed. by Costas D. Maranas, e1003580. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1003580. URL: http://dx.doi.org/10.1371/journal.pcbi.1003580.

[15] Marlon Stoeckius et al. "Simultaneous epitope and transcriptome measurement in single cells". In: *Nature Methods* 14.9 (July 2017), pp. 865–868. DOI: 10.1038/nmeth.4380. URL: https://doi.org/10.1038/nmeth.4380.

[16] Nikolai Slavov. "Scaling Up Single-Cell Proteomics". In: *Molecular and Cellular Proteomics* 21.1 (Jan. 2022), p. 100179. ISSN: 1535-9476. DOI: 10.1016/j.mcpro.2021.100179. URL: http://dx.doi.org/10.1016/j.mcpro.2021.100179.

[17] Ahmed Ali et al. "Single cell metabolism: current and future trends". In: *Metabolomics* 18.10 (Oct. 2022). ISSN: 1573-3890. DOI: 10.1007/s11306-022-01934-3. URL: http://dx.doi.org/10.1007/s11306-022-01934-3.

[18] Jiarui Ding et al. "Systematic comparison of single-cell and single-nucleus RNA-sequencing methods". In: *Nature Biotechnology* 38.6 (Apr. 2020), pp. 737–746. ISSN: 1546-1696. DOI: 10.1038/s41587-020-0465-8. URL: http://dx.doi.org/10.1038/s41587-020-0465-8.

[19] Sarah Black et al. "CODEX multiplexed tissue imaging with DNA-conjugated antibodies". In: *Nature Protocols* 16.8 (July 2021), pp. 3802–3835. ISSN: 1750-2799. DOI: 10.1038/s41596-021-00556-8. URL: http://dx.doi.org/10.1038/s41596-021-00556-8.

[20] Stanley E. Lazic, Charlie J. Clarke-Williams, and Marcus R. Munafò. "What exactly is 'N' in cell culture and animal experiments?" In: *PLOS Biology* 16.4 (Apr. 2018), e2005282. ISSN: 1545-7885. DOI: 10.1371/journal.pbio.2005282. URL: http://dx.doi.org/10.1371/journal.pbio.2005282.

[21] Geoffrey Schiebinger et al. "Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming". In: *Cell* 176.4 (Feb. 2019), 928–943.e22. ISSN: 0092-8674. DOI: 10.1016/j.cell.2019.01.006. URL: http://dx.doi.org/10.1016/j.cell.2019.01.006.

[22] Maya Shamir et al. "SnapShot: Timescales in Cell Biology". In: *Cell* 164.6 (Mar. 2016), 1302–1302.e1. ISSN: 0092-8674. DOI: 10.1016/j.cell.2016.02.058. URL: http://dx.doi.org/10.1016/j.cell.2016.02.058.

[23] Chandler Squires et al. "Causal Structure Discovery between Clusters of Nodes Induced by Latent Factors". In: *1st Conference on Causal Learning and Reasoning, CLeaR 2022, Sequoia Conference Center, Eureka, CA, USA, 11-13 April, 2022*. Ed. by Bernhard Scholkopf, Caroline Uhler, and Kun Zhang. Vol. 177. Proceedings of Machine Learning Research. PMLR, July 2022, pp. 669–687. URL: https://proceedings.mlr.press/v177/squires22a.html.

[24] Nils Eling, Michael D. Morgan, and John C. Marioni. "Challenges in measuring and understanding biological noise". In: *Nature Reviews Genetics* 20.9 (May 2019), pp. 536–548. ISSN: 1471-0064. DOI: 10.1038/s41576-019-0130-6. URL: http://dx.doi.org/10.1038/s41576-019-0130-6.

[25] Anika Gupta et al. "Inferring gene regulation from stochastic transcriptional variation across single cells at steady state". In: *Proceedings of the National Academy of Sciences* 119.34 (Aug. 2022). ISSN: 1091-6490. DOI: 10.1073/pnas.2207392119. URL: http://dx.doi.org/10.1073/pnas.2207392119.

[26] Roman A. Zubarev. "The challenge of the proteome dynamic range and its implications for in-depth proteomics". In: *Proteomics* 13.5 (Mar. 2013), pp. 723–726. ISSN: 1615-9853. DOI: 10.1002/pmic.201200451. URL: http://dx.doi.org/10.1002/pmic.201200451.

[27] Elizabeth M. Llufrio et al. "Sorting cells alters their redox state and cellular metabolome". In: *Redox Biology* 16 (June 2018), pp. 381–387. DOI: 10.1016/j.redox.2018.03.004. URL: https://doi.org/10.1016/j.redox.2018.03.004.

[28] John Arevalo et al. "Evaluating batch correction methods for image-based cell profiling". In: (Sept. 2023). DOI: 10.1101/2023.09.15.558001. URL: http://dx.doi.org/10.1101/2023.09.15.558001.

[29] Michael Murphy, Stefanie Jegelka, and Ernest Fraenkel. "Self-supervised learning of cell type specificity from immunohistochemical images". In: *Bioinformatics* 38.Supplement 1 (June 2022), pp. i395–i403. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btac263. URL: http://dx.doi.org/10.1093/bioinformatics/btac263.

[30] Alexander Lin and Alex Lu. "Incorporating knowledge of plates in batch normalization improves generalization of deep learning for microscopy images". In: *Proceedings of the 17th Machine Learning in Computational Biology meeting*. Ed. by David A Knowles, Sara Mostafavi, and Su-In Lee. Vol. 200. Proceedings of Machine Learning Research. PMLR, Nov. 2022, pp. 74–93. URL: https://proceedings.mlr.press/v200/lin22a.html.

[31] Alex J. DeGrave, Joseph D. Janizek, and Su-In Lee. "AI for radiographic COVID-19 detection selects shortcuts over signal". In: *Nature Machine Intelligence* 3.7 (May 2021), pp. 610–619. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00338-7. URL: http://dx.doi.org/10.1038/s42256-021-00338-7.

[32] Valentine Svensson. "Droplet scRNA-seq is not zero-inflated". In: *Nature Biotechnology* 38.2 (Jan. 2020), pp. 147–150. DOI: 10.1038/s41587-019-0379-5. URL: https://doi.org/10.1038/s41587-019-0379-5.

[33] Ruochen Jiang et al. "Statistics or biology: the zero-inflation controversy about scRNA-seq data". In: *Genome Biology* 23.1 (Jan. 2022). DOI: 10.1186/s13059-022-02601-5. URL: https://doi.org/10.1186/s13059-022-02601-5.

[34] Tal Ashuach et al. "MultiVI: deep generative model for the integration of multimodal data". In: *Nature Methods* 20.8 (June 2023), pp. 1222–1231. ISSN: 1548-7105. DOI: 10.1038/s41592-023-01909-9. URL: http://dx.doi.org/10.1038/s41592-023-01909-9.

[35] Ethan D. Evans et al. "Predicting human health from biofluid-based metabolomics using machine learning". In: *Scientific Reports* 10.1 (Oct. 2020). ISSN: 2045-2322. DOI: 10.1038/s41598-020-74823-1. URL: http://dx.doi.org/10.1038/s41598-020-74823-1.

[36] Michael Murphy et al. "Efficiently predicting high resolution mass spectra with graph neural networks". In: *Proceedings of the 40th International Conference on Machine Learning*. Ed. by Andreas Krause et al. Vol. 202. Proceedings of Machine Learning Research. PMLR, July 2023, pp. 25549–25562. URL: https://proceedings.mlr.press/v202/murphy23a.html.

[37] Amy F. Chen et al. "NEAT-seq: simultaneous profiling of intra-nuclear proteins, chromatin accessibility and gene expression in single cells". In: *Nature Methods* 19.5 (May 2022), pp. 547–553. ISSN: 1548-7105. DOI: 10.1038/s41592-022-01461-y. URL: http://dx.doi.org/10.1038/s41592-022-01461-y.

[38] Yang Yang et al. "Dimensionality reduction by UMAP reinforces sample heterogeneity analysis in bulk transcriptomic data". In: *Cell Reports* 36.4 (July 2021), p. 109442. ISSN: 2211-1247. DOI: 10.1016/j.celrep.2021.109442. URL: http://dx.doi.org/10.1016/j.celrep.2021.109442.

[39] Kaiwen Wang et al. "Comparative analysis of dimension reduction methods for cytometry by time-of-flight data". In: *Nature Communications* 14.1 (Apr. 2023). ISSN: 2041-1723. DOI: 10.1038/s41467-023-37478-w. URL: http://dx.doi.org/10.1038/s41467-023-37478-w.

[40] Tomoya Tanjo et al. "Practical guide for managing large-scale human genome data in research". In: *Journal of Human Genetics* 66.1 (Oct. 2020), pp. 39–52. ISSN: 1435-232X. DOI: 10.1038/s10038-020-00862-1. URL: http://dx.doi.org/10.1038/s10038-020-00862-1.

[41] Srinivas Niranj Chandrasekaran et al. "JUMP Cell Painting dataset: morphological impact of 136,000 chemical and genetic perturbations". In: (Mar. 2023). DOI: 10.1101/2023.03.23.534023. URL: http://dx.doi.org/10.1101/2023.03.23.534023.

[42] William Falcon and The PyTorch Lightning team. *PyTorch Lightning*. Version 1.4. Mar. 2019. DOI: 10.5281/zenodo.3828935. URL: https://github.com/Lightning-AI/lightning.

[43] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome Biology* 19.1 (Feb. 2018). DOI: 10.1186/s13059-017-1382-0. URL: https://doi.org/10.1186/s13059-017-1382-0.

[44] Thibault Poinsignon et al. "Working with Omics Data: An Interdisciplinary Challenge at the Crossroads of Biology and Computer Science". In: *Neuromethods* (2023), pp. 313–330. ISSN: 1940-6045. DOI: 10.1007/978-1-0716-3195-9_10. URL: http://dx.doi.org/10.1007/978-1-0716-3195-9_10.

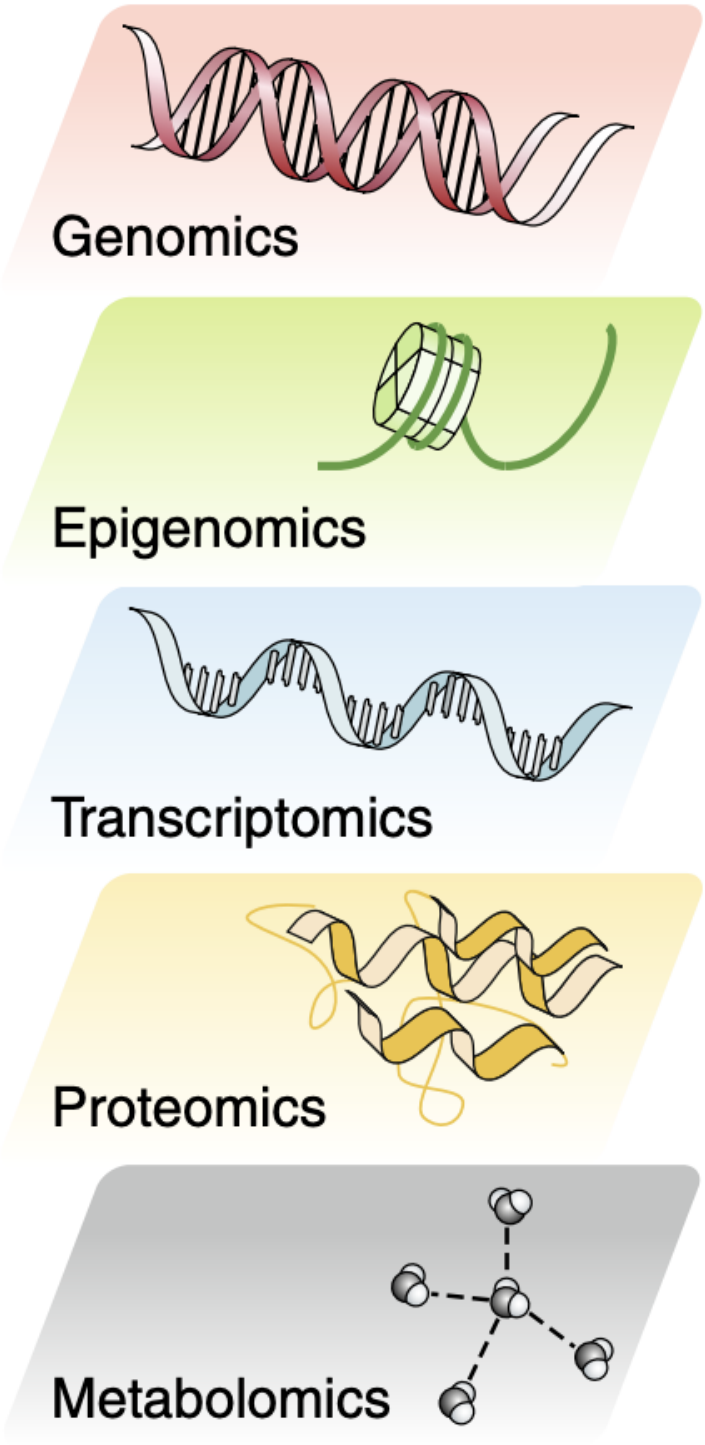

Figure 1.1: The five major classes of biological molecules measured in omics experiments. In the classical view of molecular biology, the events in each layer are caused by those in its direct predecessor, with evolution closing the loop: more recently, both low-throughput molecular biology and high-throughput omics technologies find extensive bidirectional interactions between all molecular layers. (Figure from [6].)

Chapter 2

Self-supervised learning of cell type specificity from immunohistochemical images

Authors: Michael Murphy, Stefanie Jegelka, Ernest Fraenkel

This paper was presented at ISMB 2022 and subsequently published in Bioinformatics. I conceived the idea, implemented the algorithms, carried out the analysis, and wrote the manuscript. Stefanie Jegelka and Ernest Fraenkel provided input throughout and commentary on the manuscript.

2.1 Abstract

Motivation: Advances in bioimaging now permit in-situ proteomic characterization of cell-cell interactions in complex tissues, with important applications across a spectrum of biological problems from development to disease. These methods depend on selection of antibodies targeting proteins that are expressed specifically in particular cell types. Candidate marker proteins are often identified from single-cell transcriptomic data, with variable rates of success, in part due to divergence between expression levels of proteins and the genes that encode them. In principle, marker identification could be improved by using existing databases of immunohistochemistry for thousands of antibodies in human tissue, such as the Human Protein Atlas. However, these data lack detailed annotations of the types of cells in each image.
Results: We develop a method to predict cell type specificity of protein markers from unlabeled images. We train a convolutional neural network with a self-supervised objective to generate embeddings of the images. Using nonlinear dimensionality reduction, we observe that the model clusters images according to cell types and anatomical regions for which the stained proteins are specific. We then use estimates of cell type specificity derived from an independent single-cell transcriptomics dataset to train an image classifier, without requiring any human labelling of images. Our scheme demonstrates superior classification of known proteomic markers in kidney compared to selection via single-cell transcriptomics.

2.2 Introduction

A number of technologies for multiplexed antibody-based tissue imaging have been developed in the past few years. These permit in-situ characterization of cell-to-cell surface interactions and their intracellular proteomic correlates [1, 2, 3, 4, 5], at high spatial resolution, via cyclic staining of the sample with antibodies conjugated to fluorophores [4], nucleotide barcodes [1], or metallic tags [5]. A key consideration in such experiments is the selection of an antibody panel, which can be a difficult and lengthy process. A basic criterion for this panel is to incorporate antibodies for marker proteins specific to each cell type of interest in the tissue under investigation. While suitable antibodies are readily available for certain well-studied cell types [6], large-scale efforts such as the Human Cell Atlas [7] promise to identify rarer cell types in the human body, for which selection of reliable antibody markers of cell type becomes an important and nontrivial consideration in experimental design.

Single-cell transcriptomics can be used to resolve the cell types comprising a tissue sample and their transcriptional profiles [8]. A number of recent efforts [9, 10, 11] study the problem of identifying cell type-specific marker genes from transcriptomic data. However, direct application of such approaches to the proteomic context assumes the availability of antibodies targeting the proteins of the selected marker genes, and that such antibodies have been validated in the tissue of interest. This is a strong constraint: while the literature continues to grow, Lin, Fallahi-Sichani, and Sorger [4] currently list only 257 antibodies demonstrated to work reliably with their approach [6].

Furthermore, even if a high-quality, validated antibody is available targeting a marker gene discovered from single-cell RNA sequencing data of a particular cell type of interest, if this gene is to be a useful marker *proteomically* in the tissue of interest, its transcript and protein levels also must strongly correlate in the tissue of interest. This is not universally the case, even for marker genes [12, 13, 14].

Finally, biases in single-cell sequencing protocols can lead to undersampling of certain subpopulations of cells, which negatively affects the ability to detect marker genes via differential expression [15, 16].

2.2.1 Human Protein Atlas

The Human Protein Atlas (HPA) is a large-scale compendium of proteomic experiments, dating back to 2003 and spanning several technologies including imaging, mass spectrometry, and bulk RNA sequencing [17]. Among the datasets incorporated in the HPA is a large-scale immunohistochemistry (IHC) screen, in which tens of thousands of antibodies have been imaged in tissue microarrays representing several major organs in the human body. In the IHC protocol employed by HPA, a single antibody is imaged in each sample. The antibody stains brown wherever it binds a protein target, and a hematoxylin counterstain indicates nucleic acids in blue [18].

In principle, the HPA screen could also aid in selection of marker antibodies, if we could reliably determine the type of each imaged cell. However, the HPA screen measures only one antibody at a time. As a result, the staining pattern cannot be directly compared against canonical cell type markers in order to establish its cell type specificity. At present, resolving the staining pattern in an IHC image by cell type entails visual interpretation by a human

expert [17]. This is prohibitive to carry out at finer spatial resolution than coarse anatomical regions for large numbers of images, and is necessarily subjective in nature. Although cell type labels are provided in the HPA, they are coarser than those discernible in modern single-cell sequencing experiments: for example, most images of kidney in the HPA are only annotated for staining in "glomeruli" or "tubules", while single-cell RNAseq experiments have defined, at varying resolution, between 13 [19] to 100 [20] types and subtypes of cells in the kidney.

2.2.2 Self-supervised learning

Learning informative representations of images without human supervision has been a long-standing problem in machine learning. One approach to this problem is self-supervised learning, which trains a classification or generative model to predict some attribute of the data that can be derived without a human labeler: for example, colorizing grayscale images [21], identifying distorted copies of an image [22, 23], or imputing masked patches [24]. The representation of the data learned by the self-supervised model is then used as input to a simpler supervised learning model: very often, features extracted by the model as relevant for the self-supervised task should also be relevant for solving the supervised learning task.

Recently, self-supervised methods that employ *contrastive learning* have even outperformed supervised pre-training on large-scale image recognition tasks [25, 26]. The idea of contrastive learning is to learn a representation in which semantically similar (positive) pairs of observations (x, x^+) are placed nearby, while semantically dissimilar (negative) pairs (x, x^-) are placed far apart. This is achieved by learning an encoder f that minimizes the contrastive loss function [27]:

$$\mathbb{E}_{x,x^+,\{x_i^-\}_{i=1}^B} - \log \frac{e^{f(x)^\top f(x^+)}}{e^{f(x)^\top f(x^+)} + \sum_{i=1}^B e^{f(x)^\top f(x_i^-)}} \tag{2.1}$$

where typically a minibatch of B negative examples x_i^- is used per query x instead of just one.

Since this approach does not use any human supervision, the semantic content of an image (e.g. its class label) is not available, and (dis)similarity information must be derived automatically. Contrastive learning generates positive examples for a given x via data augmentation that preserves semantics, e.g. randomly cropping, rotating, or tinting. Negative examples are obtained by sampling the training set uniformly or by more sophisticated schemes [28, 29].

2.2.3 Contributions

In short, this work makes the following contributions:

- We show how to effectively apply self-supervised learning to immunohistochemistry images, by developing a sampling procedure that generates meaningful positive and negative examples, rather than having to engineer complex data augmentations;

- We show that the learned embeddings can be combined with labelled data from a different source – single-cell transcriptomics – to obtain a classifier of cell type specificity, without requiring human labelling of images; and
- We show applying both steps to immunohistochemistry of kidney yields a representation that clusters images according to cell type specificity, and better classifies known proteomic markers than a purely transcriptomic approach.

2.2.4 Related work

Unsupervised and self-supervised learning have been employed previously to generate biologically-informative representations of genes and proteins from high-throughput imaging data [30, 31, 32]. However, the cited works focus on immunofluorescence imaging acquired in immortalized cell lines: while such datasets can indicate subcellular localization for thousands of proteins [33], they are by nature uninformative of cell type and tissue specificity. Immunofluorescence images of single cells in culture also exhibit less sample-to-sample variability than immunohistochemistry of tissue sections from human donors [34, 35, 36]. Here, we advance contrastive learning as a means of imposing invariance to sources of variability specific to immunohistochemistry.

Supervised learning has been applied previously to the HPA immunohistochemistry dataset, also for predicting subcellular localization [37, 38, 39, 40]. Ghoshal et al. [41] train a Bayesian neural network to classify cell type specificity of proteins imaged in IHC of testis, for which they rely on a training set of images manually annotated with cell type labels. In contrast, here we demonstrate how embeddings of IHC images learned via self-supervision can be combined with independent single-cell transcriptomics to predict cell type specificity without the need for human labeling beforehand.

Others have used deep learning representations to integrate imaging with transcriptomics data: Ash et al. [42] use canonical correlation analysis of paired bulk RNAseq and autoencoder representations of H&E images to identify gene sets associated with morphological features, and Badea and Stanescu [43] use intermediate activations of a classifier for the same problem. While our procedure also exploits correlation of morphology and gene expression, the problem we address in this paper is fundamentally different: we seek to establish cell type specificities of proteins to facilitate antibody selection in experimental design, while the aforementioned are concerned with linking transcriptional programs and morphological phenotypes.

2.3 Materials and Methods

2.3.1 HPA immunohistochemistry

The Human Protein Atlas includes approximately 7 million IHC images spanning tens of thousands of antibodies, in tissue microarrays derived from tens of major tissues [17, 18]. Each image represents a circular section of tissue between 0.6mm-2mm in diameter that has been stained with an antibody and a hematoxylin counterstain. There are typically 3 replicated images per antibody, each derived from a tissue sample from a different donor.

Each image is captured via microscope at 20x optical magnification, and provided via the HPA website at a resolution of approximately 3000×3000 pixels in medium-quality JPEG format. The gene whose product is nominally targeted by the antibody is indicated, as is its 'staining' classification: a qualitative assessment of the intensity of the antibody signal as 'high', 'medium', 'low', or 'not detected'. An anonymized identification number for the donor of the tissue sample is also provided.

Each antibody in the HPA also undergoes a validation procedure to determine its binding specificity. Antibodies with 'enhanced' validation satisfy two criteria: (1) an antibody passes *independent antibody* validation if it displays the same staining pattern as another antibody targeting a non-overlapping epitope of the same protein in at least two tissues; (2) an antibody passes *orthogonal* validation if its overall staining intensity matches expression of its nominal gene target in bulk RNASeq across at least two tissues. Both criteria are determined qualitatively by a human evaluator. In principle, it is unlikely for an antibody to satisfy both of these criteria yet bind to something other than its nominal target [44].

In this work, due to storage and processing limitations we restrict our scope to IHC images from version 21 of the HPA taken in a single tissue: healthy kidney. Kidney was selected in particular because it is a complex organ that displays substantial anatomical and cellular diversity [45] and with which the authors of this paper are familiar. We further filter these to only include images of antibodies that passed 'enhanced' validation, that display either 'medium' or 'high' stain intensity, and that nominally target exactly one gene. This yields a training set comprising 10,164 images of immunostained kidney sections covering 2,106 genes. Due to practical storage and memory limitations we also downsample these images to 512×512 pixels. This averages out finer-scale detail, but as we later demonstrate, still suffices to yield an informative embedding. We nonetheless anticipate our results will improve as image resolution is improved and the model is scaled accordingly.

2.3.2 Contrastively learning representations of immunohistochemical images

Contrastive learning is an effective approach for generating representations of images, both for visualization via dimensionality reduction and downstream classification tasks. This effectiveness depends on having meaningful data augmentations to generate positive examples. Simple image transformations that mimic natural variations in pose and lighting are sufficient for successful applications of contrastive learning to large datasets of natural images [46, 47, 25]. However, semantically equivalent biological images (e.g. IHC images of different markers of the same cell type) can display much more complex variability. Examples include morphological differences across tissue donors (due to e.g. age, sex, disease status), as well as technical artefacts (tearing or folding during tissue preparation, stain discoloration across batches) – all of which pose challenges for machine learning analysis of histological images generally [48, 36, 49]. Figure 2.1 provides an example of such variability among semantically equivalent images in the HPA.

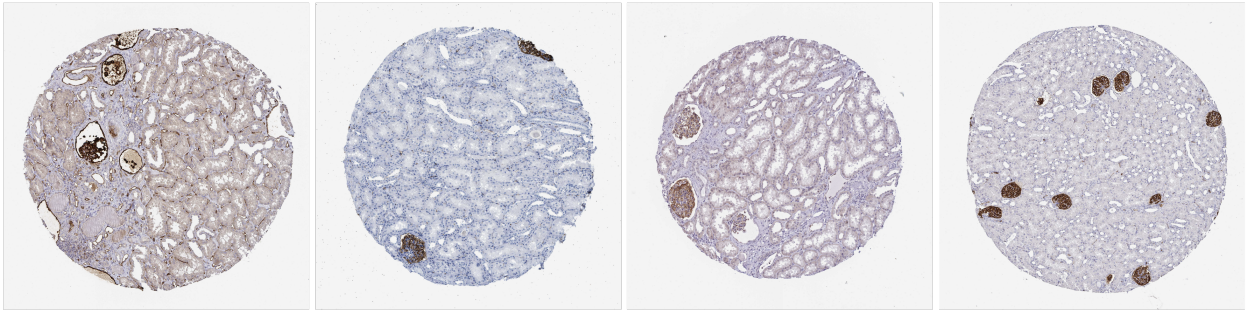

Figure 2.1: Semantically equivalent IHC images (here all of antibodies targeting the gene PODXL) display substantial variation in morphology and staining. Such artefacts would be difficult to capture via data augmentation.

Augmentations capturing variability of the sort shown in Figure 2.1 would be difficult to engineer. Hence, instead of explicitly identifying and modeling such sources of variation, we couple the contrastive learning algorithm SimCLR [46] to a biologically-informed scheme we develop for sampling positive and negative examples. We observe that the HPA already provides multiple semantically equivalent views at the level of *genes*: in addition to having multiple biologically-replicated images per antibody of tissue samples from different donors, we also often have images of multiple antibodies that target the same gene. Therefore, rather than individual images, we treat the gene as the 'observational unit' in our data: for each gene, we sample a positive pair of images from the 'equivalence class' defined by all the images of antibodies for which that gene is the nominal target. To further increase the diversity of our training set, following this sampling procedure we also apply standard image augmentations to each member of the positive pair independently: specifically, we randomly crop each image to a 256×256 patch, then randomly apply HSV color jitter, scaling, and rotation.

However, the image embeddings resulting from this approach displayed substantial clustering according to the donors from which they were derived. We believe this arises because (as we show in Figure S1 of the supplement) the donors do not co-occur uniformly across genes: the resulting imbalance in the positive pairs biases the encoder to push together images from donors that co-occur more frequently, giving rise to this (undesirable) clustering structure. We design the sampling of negative pairs to counteract this effect, noting that modified schemes for negative sampling are effective in contrastively learning debiased representations [29] and promoting inter-class separation [28]. Here, rather than sampling uniformly, we only draw negative examples from the same donor: this steers the encoder to pull such clusters apart. We also provide further exposition and quantitative evaluation of the different sampling methods in Appendix A of the supplement.

Rather than implementing a complicated scheme to generate stratified minibatches, we simply implement this by masking negative pairs from different donors within the minibatch when computing the contrastive loss. Defining the normalization term $B' = \sum_{i=1}^{B} 1_{d=d_i}$ to be the number of negative examples sampled from the same donor d as image x, we use the

following expression for the summand in equation (2.1):

$$-\log \frac{e^{f(x)^\top f(x^+)}}{e^{f(x)^\top f(x^+)} + (B/B') \sum_{i=1}^{B} 1_{d=d_i} e^{f(x)^\top f(x_i^-)}} \tag{2.2}$$

Algorithm ?? shows pseudocode of our entire sampling procedure.

Algorithm 1 SimCLR with our modified sampling procedure.

Require: batch size B, temperature τ, encoder network f, projection network ϕ, set of augmentations $\mathcal{T}$, partition of dataset into genes $\mathcal{X}_g$

while termination criterion is not met **do**

sample minibatch of genes $\{g_i\}_{i=1}^{B}$

for all $i \in 1, \ldots, B$ **do**

draw two images with donor labels $(x, d) \sim \mathcal{X}_{g_i}, (x', d') \sim \mathcal{X}_{g_i}$

draw two augmentation functions $t \sim \mathcal{T}, t' \sim \mathcal{T}$

$\tilde{x}_i, \tilde{x}'_i = t(x_i), t'(x'_i)$ # augmentation

$h_i, h'_i = f(\tilde{x}_i), f(\tilde{x}'_i)$ # representation

$z_i, z'_i = \phi(h_i), \phi(h'_i)$ # projection

end for

for all $i \in 1, \ldots, B$ and $j \in 1, \ldots, B$ **do**

$s_{i,j} = z_i^T z'_j / (\|z_i\| \|z'_j\|)$ # pairwise similarity

end for

for all $i \in 1, \ldots, B$ **do**

$B_i = \sum_{k=1}^{B} \mathbb{1}_{d_i = d_k}$ # normalization factor

$l_i = -\log \frac{\exp(s_{i,i}/\tau)}{\exp(s_{i,i}/\tau) + (B/B_i) \sum_{k \neq i} \mathbb{1}_{d_i = d_k} \exp(s_{i,k}/\tau)}$

end for

update networks f and ϕ to minimize $\mathcal{L} = \frac{1}{B} \sum_{i=1}^{B} l_i$

end while

return encoder network f

2.3.3 Encoder architecture and training

As our encoder network f we use a DenseNet-121 [50] convolutional neural network, which is initialized from weights pre-trained on ImageNet [51]. We use this architecture because it was selected by the top performer in the HPA subcellular localization challenge of Ouyang, Winsnes, Hjelmare, et al. [33]. We pass 256×256 RGB image patches into this encoder, which transforms them to an 8×8 grid of 1024-dimensional embeddings. We then average-pool this grid to a single 1024-dimensional vector, and linearly transform it to a $D = 128$-dimensional real-valued embedding h. The projection head ϕ [46] then applies a ReLU nonlinearity, followed by a second 128d linear transformation, and normalization to the unit L_2-ball. This yields a vector z on which we compute the contrastive loss. We use a temperature parameter of $\tau = 1.0$.

The model is implemented in PyTorch Lightning [52] using the Kornia [53] library for data augmentation. We fit this model using the Adam optimizer [54] for 1000 epochs, with

learning rate 5×10^{-4} and batches of size 150, using 4 nVidia Volta V100 GPUs with 32GB VRAM. Larger batch sizes did not fit in memory, even with the aid of PyTorch Lightning's automatic half-precision casting, which we employ here. To avoid arithmetic underflow we compute the contrastive loss at full 32-bit precision.

2.3.4 Learning cell type specificity from auxiliary transcriptomic labels

We next demonstrate how our embeddings can be used to classify IHC images according to cell type specificity of the stained protein. As we generally lack human labels of cell type specificity for these images, we instead train a classifier using auxiliary labels of specificity, derived from an independent single-cell transcriptomics dataset. Because transcription imperfectly correlates with protein expression, these labels act only as a noisy proxy of proteomic specificity. Therefore we should not seek a complex model that achieves very low error – as this would likely capture noise that is unrelated to protein expression. Rather, by restricting to only simple functions of embedded protein images, we prevent the classifier from learning such noise, and steer it toward what we actually desire: predictions of proteomic specificities. In this work we select a linear classifier for this task.

As our single-cell transcriptomics dataset, we use the processed data provided by Muto et al. [19] via the *cellxgene* portal [55]. This dataset consists of single nuclei from non-tumor kidney cortex from five donors (three male, two female) sequenced using the 10x Genomics Chromium v2 platform. This yielded a matrix of 19,985 cells × 22,127 genes. Muto et al. [19] performed unsupervised clustering of this data using the Louvain community-detection algorithm, and assigned a cell type to each cluster based on expression of known lineage-specific markers, resulting in 13 different cell types. The dataset provided on *cellxgene* also maps the shorthand names used by Muto et al. [19] for these cell types to terms in the Cell Ontology [56], which we use in this paper.

The learning problem is as follows: we first compute, for each gene g, its mean transcriptional expression in each of the K cell types annotated in the single-cell dataset. We then normalize this to a vector $y_g \in \Delta_K$, and train a linear classifier to predict y_g from its corresponding embedded images $\{h_i \in \mathbb{R}^D, i \in 1 \dots N : g_i = g\}$:

$$\min_{A \in \mathbb{R}^{D \times K}} - \sum_{i=1}^{N} y_{g_i}^\top \log \left(\text{softmax}(h_i^\top A) \right) \tag{2.3}$$

where log is applied element-wise.

We fit the model in equation (2.3) to the same training set as before, minus 633 images corresponding to a test set of marker genes indicated in [57]. We train for 1000 epochs using the Adam optimizer with learning rate 0.01.

2.3.5 Evaluating predictions of proteomic specificity

As we lack large-scale annotation of cell type specificity for proteins in kidney, we instead assess our model's predictions at the scale of coarser anatomical regions, and compare these to estimates of regional specificity derived solely from single-cell transcriptomic data.

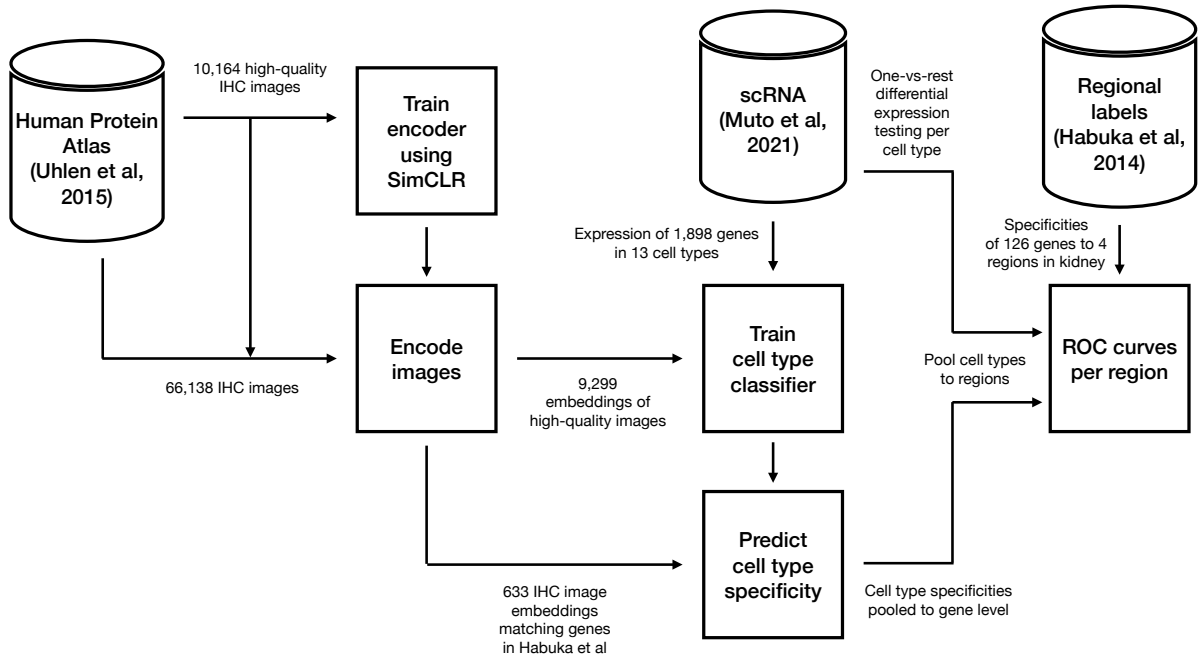

Figure 2.2: We first train an encoder on 10,164 IHC images from the HPA that pass quality filtering, and compute embeddings for all 66,138 images, which we subsequently hold fixed. We then compute mean expression in each of 13 cell types for each gene in Muto et al. [19]. We fit a linear classifier to embeddings of 9,299 high-quality images that (a) correspond to 11,657 genes in Muto et al, and (b) are not listed in Habuka, Fagerberg, Hallström, et al. [57]. We then evaluate this linear classifier on embeddings of 633 held-out HPA images matching 126 genes annotated as regional markers in Habuka, Fagerberg, Hallström, et al. [57]. We aggregate along the image axis to yield gene-level predictions, and along the cell-type axis to yield predictions for 4 anatomical regions. This permits us to compute ROC curves using the labels of Habuka, Fagerberg, Hallström, et al. [57] and evaluate against baselines that rely upon differential expression of transcripts between cell types to nominate markers.

We employ a list of marker genes identified as proteomically specific to particular kidney anatomical regions by Habuka, Fagerberg, Hallström, et al. [57]. These markers were identified first by preselecting genes with kidney-specific expression in bulk RNAseq, and then labelled with one of four anatomical regions of kidney cortex by manual inspection of IHC images from an earlier version of the HPA. Of these markers, 126 are present both in version 21 of the HPA and in the single-cell transcriptomics dataset, and we limit our evaluation to these. We hold out the corresponding images of these genes during training of the classifier.

The anatomical regions labelled in Habuka, Fagerberg, Hallström, et al. [57] each (with the exception of proximal tubules) contain multiple transcriptionally-defined cell types. As our model is trained to predict cell types, we map cell types in [19] to anatomical regions in [57] according to the scheme shown in Figure 2.4, omitting two cell types ('leukocyte' and 'fibroblast') from our analysis that lack a clear correspondence to a single anatomical region. We convert our model's predictions from cell types to regions by summing the

softmax probabilities for all cell types mapped to a given region, and then renormalizing. This yields, for each image, a vector of predicted probabilities for each of the four regions. But as our ground-truth labeling is only provided for genes (not for individual images), we generate predictions at the gene level by taking the average of all such vectors for a given gene.

As baselines, we consider assigning genes to cell types using the single-cell transcriptomics data alone. For each gene, we perform differential expression testing for each cell type, and use the test statistic as a score of cell type specificity. We then convert these scores per cell type into anatomical regions by taking the maximum test statistic over all cell types mapped to a given region. We do this separately for one-versus-rest T-tests and Wilcoxon tests (using ScanPy's [58] `sc.tl.rank_genes_groups` function), which are common practice for marker detection in scRNA analysis [59].

We also compare against a published state-of-the-art algorithm for marker detection, COMET [10]. We run COMET using default parameters and consider only test statistics of positive singleton markers in our evaluation (as Habuka, Fagerberg, Hallström, et al. [57] only specify labels for these).

We provide a schematic in Figure 2.2 that summarizes our training and evaluation workflow.

2.4 Results

2.4.1 Contrastively-learned embedding structure reflects cell type specificity of immunohistochemical stains

We first use ScanPy's UMAP dimensionality reduction [60] to visualize the embeddings learned by our model. We also cluster these embeddings using Scanpy's implementation of the Leiden community-detection algorithm (resolution parameter 0.2). This is shown in Figure 2.4, where we also color and size each image embedding according to the cell type in which the corresponding gene is most expressed, per scRNA from Muto et al. [19]. This reveals a number of clusters exhibiting specificity for particular cell types and anatomical regions in the kidney. Importantly, this structure arises solely from our self-supervised objective and without human supervision: at no point is cell type specificity of an image ever provided as a label to our algorithm.

Exemplars from these clusters are shown in Figure 2.3. We identify these exemplars by training an SVM (scikit-learn [61], RBF kernel, default parameters, 5-fold CV) to predict each embedding's cluster label, extracting the top 5 highest-scoring images for each class.

Clusters 1 and 11 are enriched in genes displaying 'epithelial cell of proximal tubule'-specific expression (hypergeometric test, $p < 10^{-202}$ and $p < 10^{-50}$ respectively).

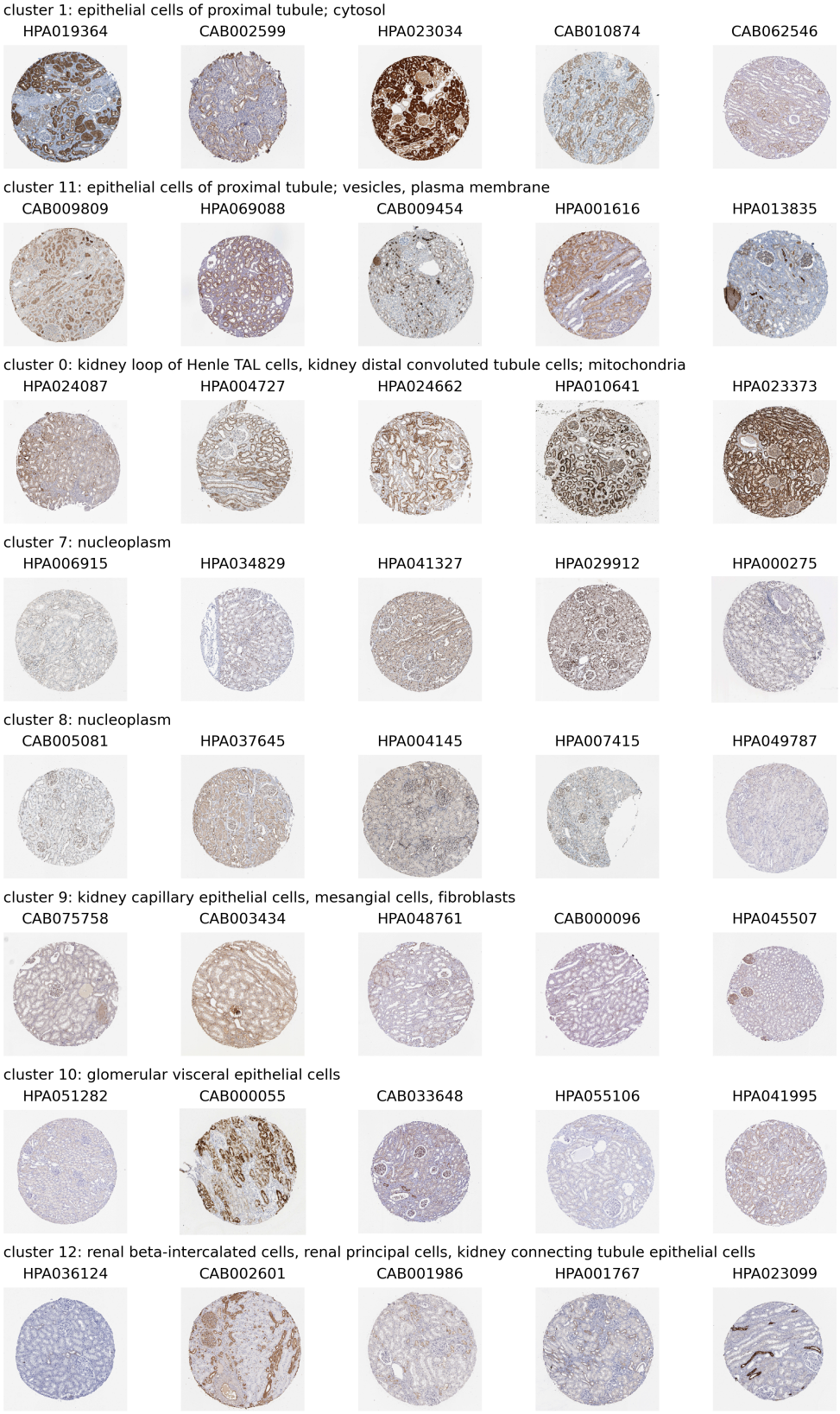

Figure 2.3: Exemplar images chosen from a subset of Leiden clusters with clear specificity for kidney anatomical regions and subcellular localizations, with HPA antibody codes.

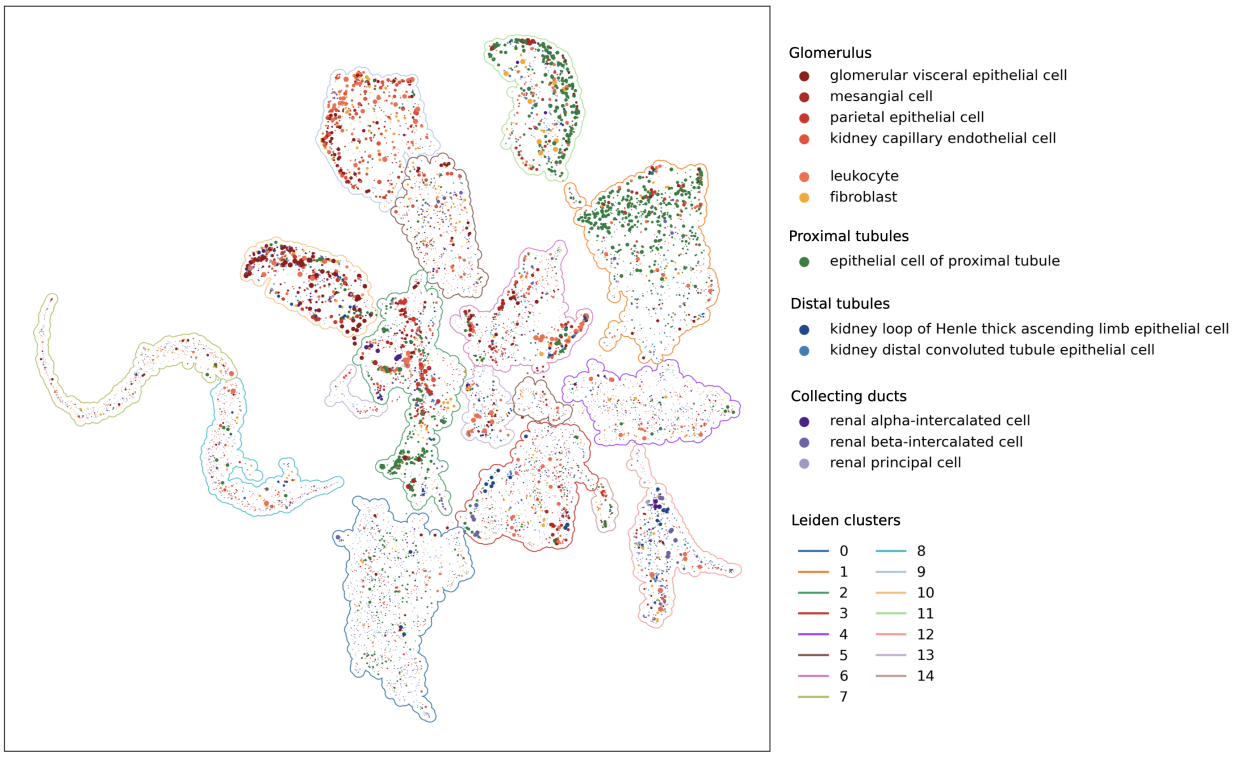

Figure 2.4: Visualization of 10,164 image embeddings via UMAP dimensionality reduction demonstrates that our method captures biological semantics of IHC images. Each dot represents an embedding of a single IHC image. We color the embedding according to the cell type in which its corresponding gene is most expressed, per the scRNA dataset of Muto et al. [19], and size each according to cell type specificity (i.e. $\arg\max_k y_{gk}$ and $\max_k y_{gk}$ respectively). Cell types are grouped according to anatomical regions in Habuka, Fagerberg, Hallström, et al. [57]; 'leukocyte' and 'fibroblast' do not correspond to any single region. Leiden clusters are numbered and outlined.

While the reason for images of this cell type separating into two different clusters is unclear to the unaided eye, hypergeometric testing for subcellular localization annotations of the respective genes [62] indicates cluster 1 primarily enriches for localization in the cytosol ($p < 10^{-44}$), while cluster 11 enriches for both vesicles ($p < 10^{-7}$) and plasma membrane ($p < 10^{-8}$).

This suggests our self-supervised approach may also be sensitive to subcellular localization, confirming findings from the supervised setting [37]. Indeed, the images in cluster 0, which enrich for loop of Henle ($p < 10^{-21}$) and distal tubule ($p < 10^{-14}$) cells, are also strongly enriched for mitochondrial localization ($p < 10^{-300}$). Similarly, clusters 7 and 8 enrich for nucleoplasmic localization ($p < 10^{-170}$ and $p < 10^{-152}$).

Clusters 9 and 10 both consist of genes expressed in the glomerulus. However, they display specificities for different cell types within that region: cluster 9 enriches for 'kidney capillary epithelial cells' ($p < 10^{-31}$), 'mesangial cells' ($p < 10^{-11}$) and 'fibroblasts' ($p < 10^{-13}$), while cluster 10 enriches for 'glomerular visceral epithelial cells' ($p < 10^{-30}$).

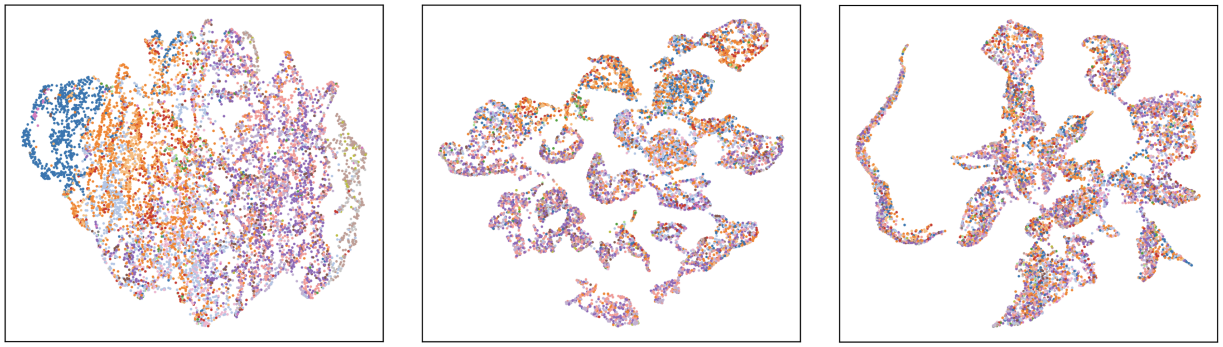

Figure 2.5: UMAP plots of image embeddings learned from three sampling schemes: image augmentations only (left); + grouping positive examples by gene (middle); + stratifying negatives by donor (right). Color indicates one of 18 anonymized donor labels. The last scheme leads to visibly better mixing of donors across clusters.

Another cluster with interpretable anatomical specificity is 12, which enriches for genes specific to 'renal beta-intercalated cells' ($p < 10^{-16}$) and 'renal principal cells' ($p < 10^{-18}$), both of which are found in the collecting duct of the kidney. The cluster is also enriched for 'kidney connecting tubule epithelial cells' ($p < 10^{-22}$), which physically link the distal tubule to the connecting duct [63].

2.4.2 Biologically-informed sampling captures semantic content and imparts invariance to donor identity

Figure 2.5 displays the effect of our positive and negative sampling procedures via UMAP visualizations of the resulting embeddings.

We first consider a self-supervised encoder trained using only standard image augmentations (left). This does not result in a visualization with clear clustering structure, and images tend to colocate with other images derived from the same donor. In comparison, our approach of sampling positive examples from other images of the same gene (middle) imparts clearly visible global clustering structure.

However, there remain some clusters exhibiting visible imbalance with respect to the donor label. This is fixed when we additionally restrict negative examples to images derived from the same donor (right). We quantify invariance of our representation to the identity of the tissue donor via 5-fold cross validation accuracy of a logistic regression trained to predict the donor label from the embedding. When sampling negatives uniformly, the donor label can be accurately classified 39.7% of the time; our negative sampling procedure makes this task more difficult, reducing to 29.5% accuracy.

2.4.3 Immunohistochemical classification yields superior predictions of regional specificity over transcriptomics

We evaluate our model's performance in classifying images of proteomic markers for the four anatomical regions specified in Habuka, Fagerberg, Hallström, et al. [57], against the

common practice of one-versus-rest T-tests and Wilcoxon tests [59] as well as the state-of-the-art method COMET [10]. Figure 2.6 shows one-versus-rest receiver operator characteristic curves for each region.

Our approach incorporating immunohistochemistry demonstrates superior accuracy in proximal tubule (ΔAUC = 0.061), glomerulus (ΔAUC = 0.191), collecting duct (ΔAUC = 0.106), and comparable accuracy in distal tubules (ΔAUC = -0.011) over the best transcriptional baseline for each region.

We provide two examples of markers for which immunohistochemical and transcriptomic predictions disagree. Figure 2.7 shows a selected image from each, along with the corresponding gene expression per each cell type in Muto et al. [19].

Habuka, Fagerberg, Hallström, et al. [57] label PTH2R as a proteomic marker of cells in the glomerulus. Our model agrees, ranking the antibody HPA010655 for PTH2R third-most-specific to glomeruli of the 217 antibodies targeting genes in Habuka, Fagerberg, Hallström, et al. [57]. In comparison, the overall best-performing scRNA baseline (T-test) ranks PTH2R *last* among the 126 marker genes by glomerular specificity, instead assigning it to proximal tubules (ranked 8$^{\text{th}}$ of 126).

We suggest an explanation for the misclassification based on scRNA: using in-situ hybridization, Usdin et al. [64] observe PTH2R is specifically transcribed in a small subpopulation of glomerular cells in rat kidney. It is possible that biases in the scRNA sequencing or analysis protocols negatively affected detection of that minority cell type [16, 15]. This would present a potential mode of failure when relying upon single-cell transcriptomics data to determine proteomic specificity.

Habuka, Fagerberg, Hallström, et al. [57] label SLC2A9 as a marker of proximal tubules. The best transcriptomic baseline accurately classifies this gene, ranking it 6$^{\text{th}}$ of 126 in proximal tubule specificity. Conversely, our method incorrectly classifies this gene's antibody, HPA066229, as most specific to distal tubules (ranked 25$^{\text{th}}$ of 217, versus 187$^{\text{th}}$ for proximal tubules).

Rather than a mistake of our classifier, we suggest this may be a limitation of relying on immunohistochemistry: visual inspection of the images of that gene also indicates staining of distal tubules, highlighted with red arrows in Figure 2.7. However, So and Thorens [65] confirm Habuka, Fagerberg, Hallström, et al. [57]'s annotation, indicating SLC2A9 is known to selectively express in proximal tubules. This may indicate off-target binding of the antibody: while Habuka, Fagerberg, Hallström, et al. [57] may have observed and accounted for this in their annotation, our approach cannot discern such cases.

We also provide in Appendix B of the supplement a demonstration of our method applied to IHC images and scRNA of testis. We benchmark against transcriptomic baselines as shown here, as well as DeepHistoClass [41], a *supervised* learning algorithm trained on images of testis that were manually labelled for cell-type specificity by human experts.

2.5 Discussion

In this paper, we develop a contrastive learning algorithm for learning representations of immunohistochemistry images in the Human Protein Atlas. We demonstrate our approach to sampling positive and negative examples leads to a representation that captures biolog-

ical semantics of IHC images, without needing to engineer complex data augmentations. When applied to images of the kidney, the resulting embeddings yield clusters that display specificity for different cell types and anatomical regions. We then incorporate auxiliary labels from an independent single-cell transcriptomics dataset to train an image classifier that predicts cell type specificity without requiring human annotation of those cell types. This classifier better recovers known proteomic markers than prioritization solely via differential transcriptional expression.

One important potential application of our method is toward designing multiplexed spatial proteomics experiments, for which antibody panel selection is an important consideration. The ability to screen candidate marker genes beforehand for proteomic specificity, as opposed to solely transcriptomic specificity, should save time during validation of such panels.

While we only consider images of kidney here, our contrastive learning procedure is applicable to any tissue represented in the Human Protein Atlas, and the embeddings can be used for any prediction or visualization task for which invariances of the sort we learn are desirable. We also point out that while training the model does necessitate the unique scale of the HPA specifically, in principle it can be subsequently applied to any IHC image acquired in a tissue that was represented in the training set. Our subsequent classification step can also use any data type that associates genes or proteins with cell type specificity: it is not limited to our particular choice of single-cell transcriptomics dataset, nor to the granularity of cell type definitions therein. It therefore will benefit from efforts to discover and catalog finer distinctions between cell types, such as the Human Cell Atlas [7].

In addition to cell type specificity, we also observe our approach is sensitive to subcellular localization of markers. This result is unsurprising, as multiple previous works demonstrate effective prediction of subcellular localization from IHC images in the supervised context [37, 38, 39, 40]: here we find such information can also be detected without supervision. A representation that disentangles these two modes of specificity would be a promising direction of future work.

We note some limitations with our method in its current implementation. We had to downsample the high-resolution images in the HPA substantially to accommodate our choice of architecture, and GPU memory constraints prevented us from using larger batch sizes as recommended for SimCLR [46]. We expect improvements in memory-efficient contrastive learning such as He, Fan, Wu, et al. [25] will permit us to use HPA images at their native resolution, which should yield finer distinctions between cell types and subcellular compartments. We also anticipate multi-scale attention [66] will be particularly useful for distinguishing cell type markers by finer-scale features.

Finally, our approach is fundamentally limited by antibody cross-reactivity: while we filter our training set using the HPA's reliability criteria to counteract this, in principle an approach that attempts to predict cell type specificity of proteins from immunohistochemical images will fail when antibodies bind to proteins other than their nominal targets. On the other hand, genes that exhibit systematic disagreement between immunohistochemical and transcriptomic estimates of cell type specificity across multiple antibodies and tissues may present interesting candidates for biological followup in their own right.

Acknowledgements

The authors thank the reviewers for their feedback, and acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing HPC resources that have contributed to the research results reported within this paper. MM is supported by an NSERC Alexander Graham Bell Canadian Graduate Scholarship and by funding from the Chan-Zuckerberg Initiative.

2.6 Appendix

2.6.1 Comparison of sampling procedures

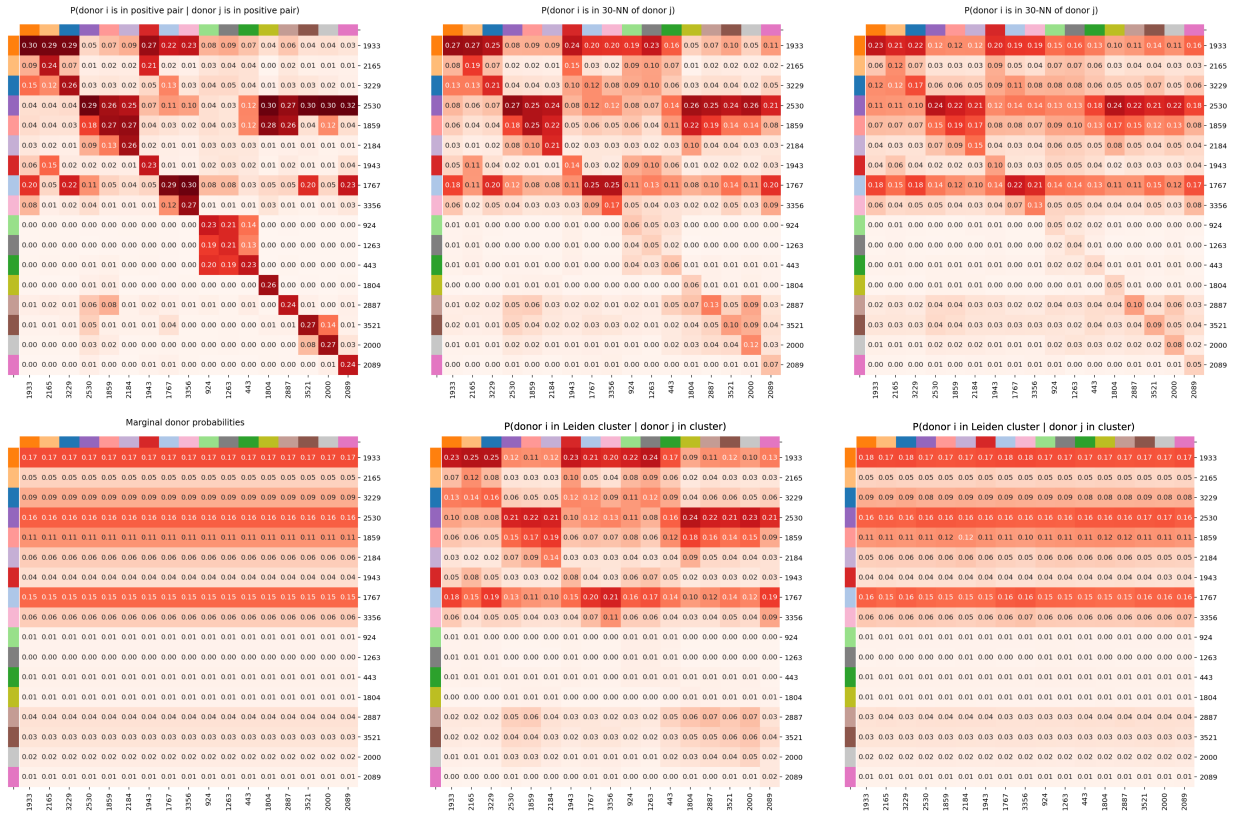

Figure 2.8: Top left, conditional probability of drawing a positive example from donor i for an image from donor j; top middle, conditional probability of an embedding of an image from donor i lying in the 30-nearest-neighbors of an embedding from donor j without our negative sampling procedure; top right, the same, with our negative sampling procedure; bottom left; marginal probability of drawing an example from donor i; bottom middle and bottom right, same as top but with Leiden clusters.

Figure 2.8 suggests the co-occurrence of donors among positive pairs is reflected in the geometry of the learned embeddings: specifically, the conditional probabilities of observing

Sampling method	AUC-P	AUC-G	AUC-D	AUC-C	AUC	Accuracy
Image augmentation	0.957	0.941	0.798	0.883	0.895	0.621
Image augmentation + sampling at gene level	0.972	0.954	0.877	0.950	0.939	0.397
Image augmentation + sampling at gene level + rejecting positives	**0.989**	**0.991**	0.934	0.953	**0.967**	0.464
Image augmentation + sampling at gene level + rejecting negatives	0.967	0.982	**0.949**	**0.960**	0.964	**0.295**

Table 2.1: Performance of different sampling strategies on the downstream classification task (test AUC on proximal tubule (P), glomerulus (G), distal tubule (D), collecting duct (C), and overall; higher better), and invariance to donor identity of the embeddings (5-fold CV accuracy; *lower* better).

donor i as a positive example for donor j when grouping images by genes (top left), resemble the proportion of images from donor i in the 30 Euclidean nearest neighbors of an embedding of donor j (top middle; KL divergence between these two distributions for fixed j, averaged across all $j = 0.215$). The negative sampling procedure drives these two donor-conditional probability distributions farther apart (top right; average KL divergence = 0.349). It also drives the proportion of images of donor i in a 30-NN of donor j much closer to what would arise if each donor were just represented in the neighborhood proportionally to its total number of images (bottom left; average KL divergence 0.228 before, 0.082 after).

We can similarly compute the probability of observing an image from donor i within a Leiden cluster (resolution = 0.2) of the embeddings, conditional upon donor j also being present in that cluster. While the relation to the donor co-occurrence is weaker to begin with (bottom middle), the effect of the negative sampling is even stronger at the cluster level: donors are assigned to clusters essentially randomly (bottom right).

We further note it is not possible with this dataset to directly correct the imbalance in donor pairings by instead reweighting each gene's positives: generally only a few (median 3, maximum 6) of the 18 donors are represented per gene, so most pairings of donors will occur for that gene with probability zero.

In Table 2.1 we provide quantitative metrics for the effectiveness of the different strategies for sampling positive and negative pairs: both in terms of test performance on the downstream classification task, as well as invariance of the embedding to the donor identity. As we describe in the main text, the latter is quantified via 5-fold cross validation accuracy of a logistic regression trained to predict the donor label from the embedding: a more invariant embedding will make this prediction task harder.

We also considered an alternative approach for sampling, which we include here for completeness: rather than only sampling negative pairs originating from the same donor, we only drew positive pairs from different donors. This was similarly implemented via masking at the minibatch level. This did not affect the classification accuracy of the downstream task (mean AUC = 0.967 across the four regions, versus 0.964 for negative sampling). It also achieved the opposite of our desired effect, *increasing* linear separability of donors. The

embeddings derived from this approach, as well as those of the other three we investigated, are shown in Figure 2.9.

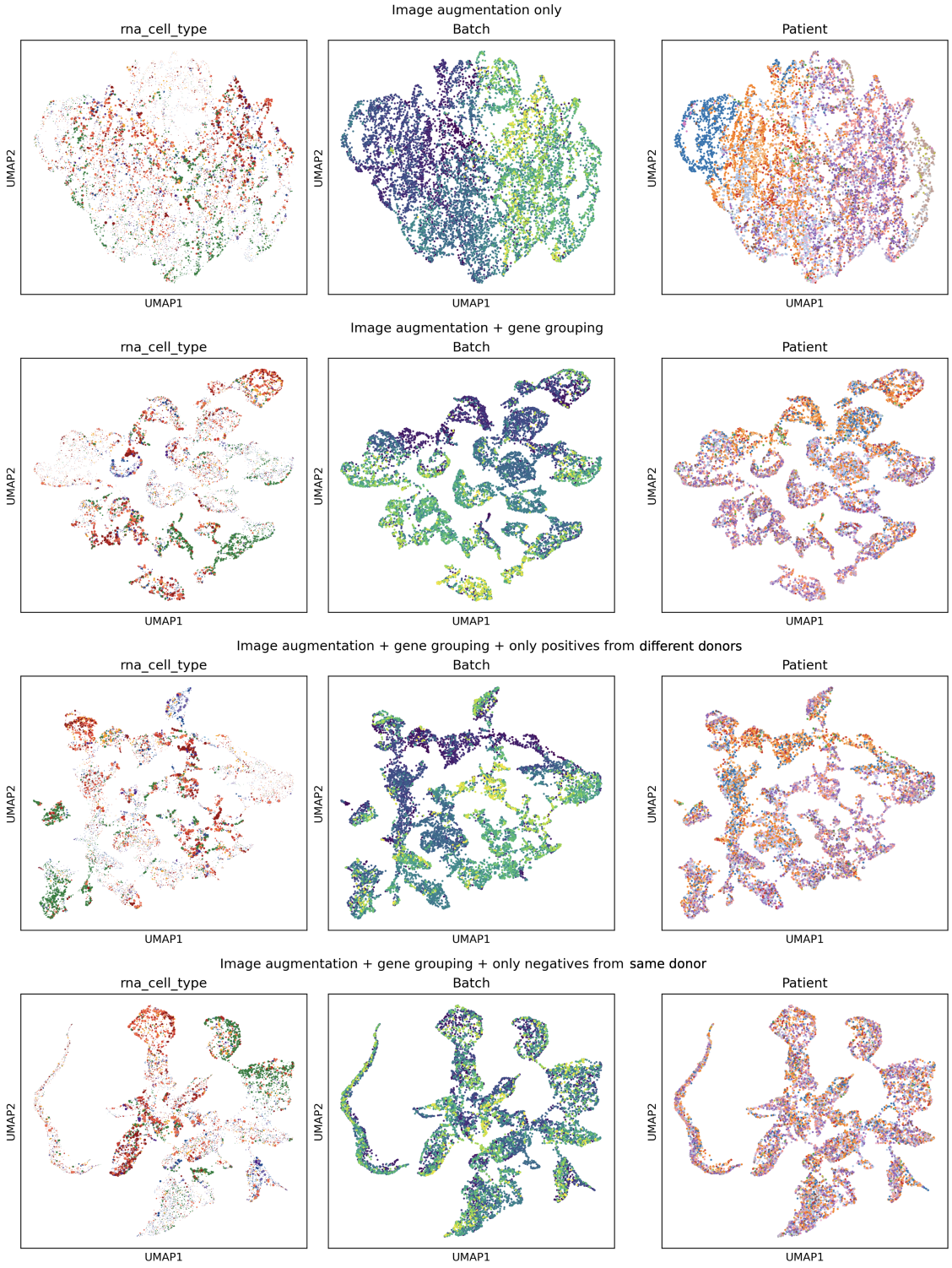

Figure 2.9: UMAP plots of IHC image embeddings from the four sampling schemes. The left column indicates the cell-type specificity of the respective gene in Muto et al. [19]. The middle column, "Batch", is an ordinal index of the individual TMA on which the image was acquired (each TMA contains three images of the same antibody, each from a different donor). It is clearly confounded with the donor label; despite only using the latter, our final scheme visibly achieves the best mixing across clusters of this label as well.

2.6.2 Evaluation on immunohistochemistry of testis

We also apply our method to a different tissue: healthy human testis. We select this tissue because [41] provide (1) a manually-annotated test set of IHC images with cell-type labels (as opposed to regional specificity), and (2) a supervised learning model, DeepHistoClass, that provides a proteomic benchmark against which we can evaluate our method.

To train our encoder, we use 4777 images of testis from version 21 of the Human Protein Atlas, selected using the same filtering criteria described in the main text for kidney. The model and training parameters are identical to the procedure described in the main text.

To train our classifier, we employ cell type specificities from the scRNA dataset of [67]. As the per-cell labels of cell type were not provided in the published data by the authors, we derive them by running ScanPy's Leiden clustering (default parameters) after log-transformation and PCA, and assign clusters to cell types via manual inspection of transcriptional marker genes described in [67]. UMAP plots labelled with these clusters are shown in Figure 2.10 and our assignments of these to the cell types in [41].

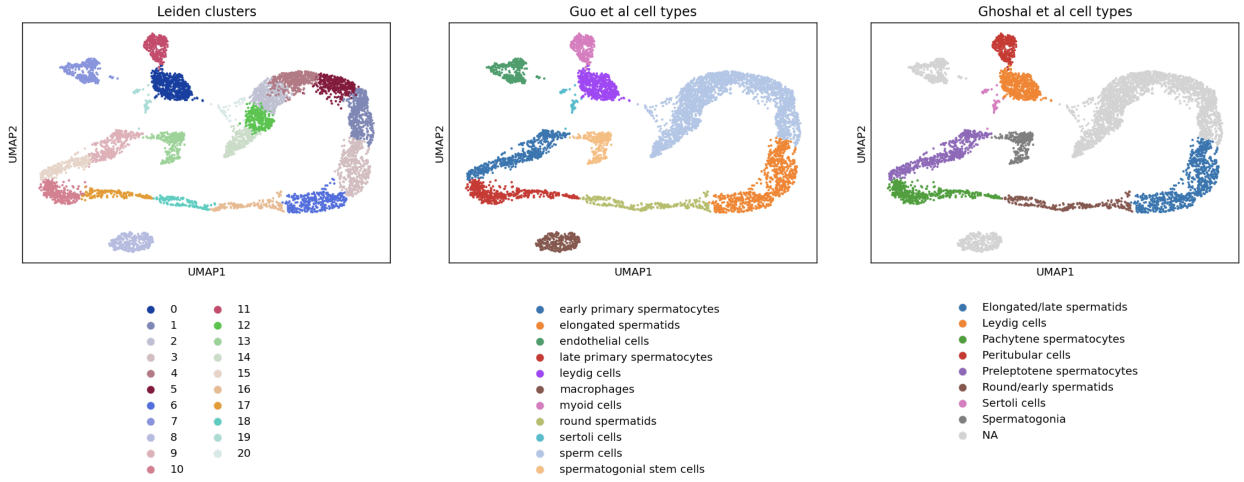

Figure 2.10: Labelled UMAP embeddings of the scRNA data in [67]. Each point represents a single cell. Cells labelled "NA" are not present among the labels in [41].

[41] provides a test set of human annotations for 1374 images from versions 18 and 19 of the Human Protein Atlas, which we also employ as a test set (after removing 31 images also present in our training set). Figure 2.11 shows the performance of their model, ours, and transcriptomic tests of differential expression using [67]. (For both image-based methods, as in the main text, we aggregate labels to gene-level via averaging over all images annotated with a given gene.)

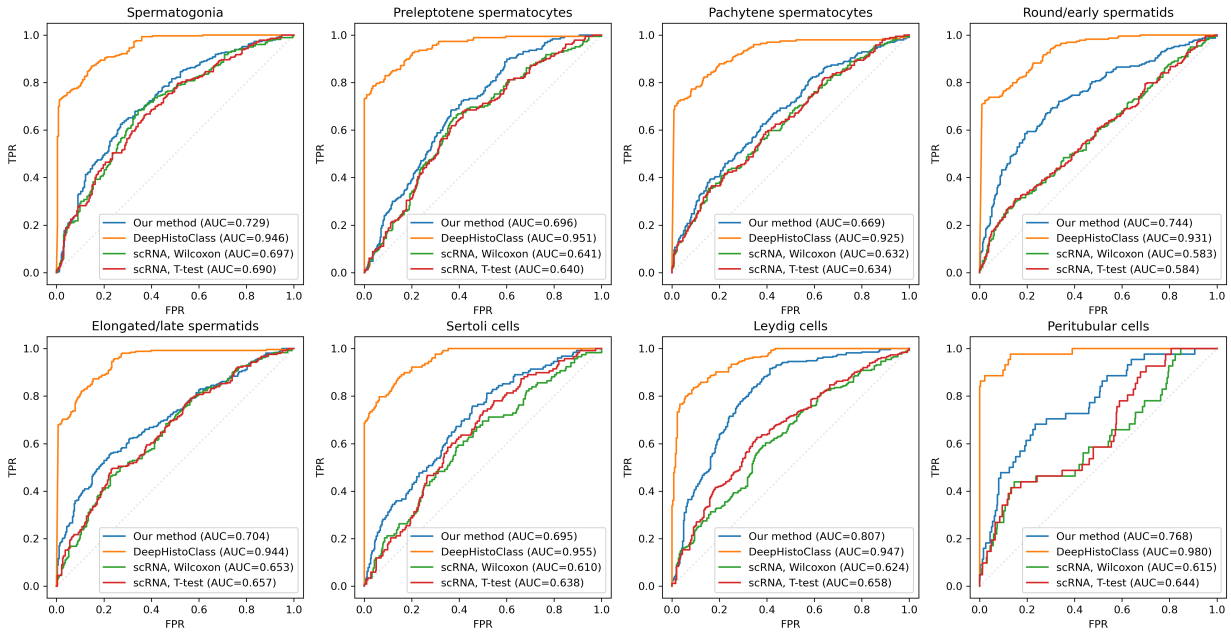

Figure 2.11: AUC curves of our method, DeepHistoClass, and two transcriptomic baselines for classifying marker genes in testis.

While we outperform the scRNA-only baselines, we observe that DeepHistoClass outperforms our approach in turn by a substantial margin. This is unsurprising. DeepHistoClass is a *supervised* learning algorithm, and therefore has access to direct labels of proteomic specificity at the level of individual images during training. By comparison, our method only has access to a noisy proxy in the form of transcriptomic specificity. The additional supervision information used by DeepHistoClass comes at a cost: Ghoshal et al. [41] required human experts to manually annotate thousands of immunohistochemical images. Our procedure is much cheaper by comparison, requiring *no human annotation of images*, and is therefore applicable even when expert annotation of cell types at that scale is impractical.

We also believe our practice of downsampling images to 512×512 adversely affected our performance on testis. While cell identity in kidney largely corresponds with macroscopic tissue organization, cells of different types in testis (such as spermatids at different developmental stages) intermix spatially. These therefore likely necessitate finer-scale information (e.g. cellular morphology, chromatin organization) to distinguish [68], which our downsampling procedure removes. In comparison, DeepHistoClass is trained on full-resolution 3000×3000 images. It is tractable to train that method at full resolution because it is intrinsically less memory intensive, being a supervised learning algorithm with only linear space complexity in batch size. SimCLR is on the other hand quadratic in this requirement [46].

References

[1] Yury Goltsev, Nikolay Samusik, Julia Kennedy-Darling, et al. "Deep Profiling of Mouse Splenic Architecture with CODEX Multiplexed Imaging". In: *Cell* 174.4 (Aug. 2018), 968–981.e15. DOI: 10.1016/j.cell.2018.07.010. URL: https://doi.org/10.1016/j.cell.2018.07.010.

[2] Christian M. Schürch, Salil S. Bhate, Graham L. Barlow, et al. "Coordinated Cellular Neighborhoods Orchestrate Antitumoral Immunity at the Colorectal Cancer Invasive Front". In: *Cell* 182.5 (Sept. 2020), 1341–1359.e19. DOI: 10.1016/j.cell.2020.07.005. URL: https://doi.org/10.1016/j.cell.2020.07.005.

[3] Gabriele Gut, Markus D. Herrmann, and Lucas Pelkmans. "Multiplexed protein maps link subcellular organization to cellular states". In: *Science* 361.6401 (Aug. 2018). DOI: 10.1126/science.aar7042. URL: https://doi.org/10.1126/science.aar7042.

[4] Jia-Ren Lin, Mohammad Fallahi-Sichani, and Peter K. Sorger. "Highly multiplexed imaging of single cells using a high-throughput cyclic immunofluorescence method". In: *Nature Communications* 6.1 (Sept. 2015). DOI: 10.1038/ncomms9390. URL: https://doi.org/10.1038/ncomms9390.

[5] Charlotte Giesen, Hao A O Wang, Denis Schapiro, et al. "Highly multiplexed imaging of tumor tissues with subcellular resolution by mass cytometry". In: *Nature Methods* 11.4 (Mar. 2014), pp. 417–422. DOI: 10.1038/nmeth.2869. URL: https://doi.org/10.1038/nmeth.2869.

[6] Laboratory of Systems Pharmacology. *Recommended Antibodies*. http://www.cycif.org/antibodies/rec Feb. 2021.

[7] Aviv Regev, Sarah A Teichmann, Eric S Lander, et al. "The Human Cell Atlas". In: *eLife* 6 (Dec. 2017). DOI: 10.7554/elife.27041. URL: https://doi.org/10.7554/elife.27041.

[8] Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, et al. "SC3: consensus clustering of single-cell RNA-seq data". In: *Nature Methods* 14.5 (Mar. 2017), pp. 483–486. DOI: 10.1038/nmeth.4236. URL: https://doi.org/10.1038/nmeth.4236.

[9] Bianca Dumitrascu et al. "Optimal marker gene selection for cell type discrimination in single cell analyses". In: *Nature Communications* 12.1 (Feb. 2021). DOI: 10.1038/s41467-021-21453-4. URL: https://doi.org/10.1038/s41467-021-21453-4.

[10] Conor Delaney, Alexandra Schnell, Louis V Cammarata, et al. "Combinatorial prediction of marker panels from single-cell transcriptomic data". In: *Molecular Systems Biology* 15.10 (Oct. 2019). DOI: 10.15252/msb.20199005. URL: https://doi.org/10.15252/msb.20199005.

[11] Yixuan Qiu et al. "Identification of cell-type-specific marker genes from co-expression patterns in tissue samples". In: *Bioinformatics* 37.19 (Apr. 2021). Ed. by Anthony Mathelier, pp. 3228–3234. DOI: 10.1093/bioinformatics/btab257. URL: https://doi.org/10.1093/bioinformatics/btab257.

[12] Marlon Stoeckius, Christoph Hafemeister, William Stephenson, et al. "Simultaneous epitope and transcriptome measurement in single cells". In: *Nature Methods* 14.9 (July 2017), pp. 865–868. DOI: 10.1038/nmeth.4380. URL: https://doi.org/10.1038/nmeth.4380.

[13] Haibiao Gong, Xiaohui Wang, Benjamin Liu, et al. "Single-cell protein-mRNA correlation analysis enabled by multiplexed dual-analyte co-detection". In: *Scientific Reports* 7.1 (June 2017). DOI: 10.1038/s41598-017-03057-5. URL: https://doi.org/10.1038/s41598-017-03057-5.

[14] Zilu Zhou et al. "Surface protein imputation from single cell transcriptomes by deep neural networks". In: *Nature Communications* 11.1 (Jan. 2020). DOI: 10.1038/s41467-020-14391-0. URL: https://doi.org/10.1038/s41467-020-14391-0.

[15] Elena Denisenko, Belinda B. Guo, Matthew Jones, et al. "Systematic assessment of tissue dissociation and storage biases in single-cell and single-nucleus RNA-seq workflows". In: *Genome Biology* 21.1 (June 2020). DOI: 10.1186/s13059-020-02048-6. URL: https://doi.org/10.1186/s13059-020-02048-6.

[16] Jiarui Ding, Xian Adiconis, Sean K. Simmons, et al. "Systematic comparison of single-cell and single-nucleus RNA-sequencing methods". In: *Nature Biotechnology* 38.6 (Apr. 2020), pp. 737–746. DOI: 10.1038/s41587-020-0465-8. URL: https://doi.org/10.1038/s41587-020-0465-8.

[17] Mathias Uhlen, Linn Fagerberg, Björn M. Hallström, et al. "Tissue-based map of the human proteome". In: *Science* 347.6220 (Jan. 2015). DOI: 10.1126/science.1260419. URL: https://doi.org/10.1126/science.1260419.

[18] Caroline Kampf et al. "Production of Tissue Microarrays, Immunohistochemistry Staining and Digitalization Within the Human Protein Atlas". In: *Journal of Visualized Experiments* 63 (May 2012). DOI: 10.3791/3620. URL: https://doi.org/10.3791/3620.

[19] Yoshiharu Muto et al. "Single cell transcriptional and chromatin accessibility profiling redefine cellular heterogeneity in the adult human kidney". In: *Nature Communications* 12.1 (Apr. 2021). DOI: 10.1038/s41467-021-22368-w. URL: https://doi.org/10.1038/s41467-021-22368-w.

[20] Blue B. Lake, Rajasree Menon, Seth Winfree, et al. "An atlas of healthy and injured cell states and niches in the human kidney". In: *bioRxiv* (2021). DOI: 10.1101/2021.07.28.454201. eprint: https://www.biorxiv.org/content/early/2021/07/29/2021.07.28.454201.full.pdf. URL: https://www.biorxiv.org/content/early/2021/07/29/2021.07.28.454201.

[21] Richard Zhang, Phillip Isola, and Alexei A. Efros. "Colorful Image Colorization". In: *Computer Vision - ECCV 2016*. Springer International Publishing, 2016, pp. 649–666. DOI: 10.1007/978-3-319-46487-9_40. URL: https://doi.org/10.1007/978-3-319-46487-9_40.

[22] Alexey Dosovitskiy et al. "Discriminative Unsupervised Feature Learning with Exemplar Convolutional Neural Networks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.9 (Sept. 2016), pp. 1734–1747. DOI: 10.1109/tpami.2015.2496141. URL: https://doi.org/10.1109/tpami.2015.2496141.

[23] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. "Unsupervised Representation Learning by Predicting Image Rotations". In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL: https://openreview.net/forum?id=S1v4N2l0-.

[24] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, et al. "Context Encoders: Feature Learning by Inpainting". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 2536–2544. DOI: 10.1109/CVPR.2016.278. URL: https://doi.org/10.1109/CVPR.2016.278.

[25] Kaiming He, Haoqi Fan, Yuxin Wu, et al. "Momentum Contrast for Unsupervised Visual Representation Learning". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 9726–9735. DOI: 10.1109/CVPR42600.2020.00975. URL: https://doi.org/10.1109/CVPR42600.2020.00975.

[26] Ishan Misra and Laurens van der Maaten. "Self-Supervised Learning of Pretext-Invariant Representations". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 6706–6716. DOI: 10.1109/CVPR42600.2020.00674. URL: https://openaccess.thecvf.com/content%5C_CVPR%5C_2020/html/Misra%5C_Self-Supervised%5C_Learning%5C_of%5C_Pretext-Invariant%5C_Representations%5C_CVPR%5C_2020%5C_paper.html.

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. "Representation Learning with Contrastive Predictive Coding". In: *CoRR* abs/1807.03748 (2018). arXiv: 1807.03748. URL: http://arxiv.org/abs/1807.03748.

[28] Joshua David Robinson et al. "Contrastive Learning with Hard Negative Samples". In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL: https://openreview.net/forum?id=CR1XOQ0UTh-.

[29] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, et al. "Debiased Contrastive Learning". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al. 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/63c3ddcc7b23daa1e42dc41f9a44a873-Abstract.html.

[30] Hirofumi Kobayashi et al. "Self-Supervised Deep Learning Encodes High-Resolution Features of Protein Subcellular Localization". In: *bioRxiv* (2021). DOI: 10.1101/2021.03.29.437595. eprint: https://www.biorxiv.org/content/early/2021/07/03/2021.03.29.437595.full.pdf. URL: https://www.biorxiv.org/content/early/2021/07/03/2021.03.29.437595.

[31] Alex X. Lu et al. "Learning unsupervised feature representations for single cell microscopy images with paired cell inpainting". In: *PLOS Computational Biology* 15.9 (Sept. 2019). Ed. by Juan Caicedo, e1007348. DOI: 10.1371/journal.pcbi.1007348. URL: https://doi.org/10.1371/journal.pcbi.1007348.

[32] Aditya Pratapa, Michael Doron, and Juan C. Caicedo. "Image-based cell phenotyping with deep learning". In: *Current Opinion in Chemical Biology* 65 (Dec. 2021), pp. 9–17. DOI: 10.1016/j.cbpa.2021.04.001. URL: https://doi.org/10.1016/j.cbpa.2021.04.001.

[33] Wei Ouyang, Casper F. Winsnes, Martin Hjelmare, et al. "Analysis of the Human Protein Atlas Image Classification competition". In: *Nature Methods* 16.12 (Nov. 2019), pp. 1254–1261. DOI: 10.1038/s41592-019-0658-6. URL: https://doi.org/10.1038/s41592-019-0658-6.

[34] Kyuseok Im et al. "An Introduction to Performing Immunofluorescence Staining". In: *Methods in Molecular Biology*. Springer New York, Dec. 2018, pp. 299–311. DOI: 10.1007/978-1-4939-8935-5_26. URL: https://doi.org/10.1007/978-1-4939-8935-5_26.

[35] Shailja Chatterjee. "Artefacts in histopathology". In: *Journal of Oral and Maxillofacial Pathology* 18.4 (2014), p. 111. DOI: 10.4103/0973-029x.141346. URL: https://doi.org/10.4103/0973-029x.141346.

[36] David Tellez et al. "Quantifying the effects of data augmentation and stain color normalization in convolutional neural networks for computational pathology". In: *Medical Image Anal.* 58 (2019). DOI: 10.1016/j.media.2019.101544. URL: https://doi.org/10.1016/j.media.2019.101544.

[37] Jin-Xian Hu et al. "Incorporating label correlations into deep neural networks to classify protein subcellular location patterns in immunohistochemistry images". In: *Proteins: Structure, Function, and Bioinformatics* (Oct. 2021). DOI: 10.1002/prot.26244. URL: https://doi.org/10.1002/prot.26244.

[38] Justin Newberg and Robert F. Murphy. "A Framework for the Automated Analysis of Subcellular Patterns in Human Protein Atlas Images". In: *Journal of Proteome Research* 7.6 (Apr. 2008), pp. 2300–2308. DOI: 10.1021/pr7007626. URL: https://doi.org/10.1021/pr7007626.

[39] Wei Long, Yang Yang, and Hong-Bin Shen. "ImPLoc: a multi-instance deep learning model for the prediction of protein subcellular localization based on immunohistochemistry images". In: *Bioinformatics* 36.7 (Dec. 2019). Ed. by Jonathan Wren, pp. 2244–2250. DOI: 10.1093/bioinformatics/btz909. URL: https://doi.org/10.1093/bioinformatics/btz909.

[40] Jieyue Li et al. "Automated Analysis and Reannotation of Subcellular Locations in Confocal Images from the Human Protein Atlas". In: *PLoS ONE* 7.11 (Nov. 2012). Ed. by Shoba Ranganathan, e50514. DOI: 10.1371/journal.pone.0050514. URL: https://doi.org/10.1371/journal.pone.0050514.

[41] Biraja Ghoshal et al. "DeepHistoClass: A Novel Strategy for Confident Classification of Immunohistochemistry Images Using Deep Learning". In: *Molecular and Cellular Proteomics* 20 (2021), p. 100140. DOI: 10.1016/j.mcpro.2021.100140. URL: https://doi.org/10.1016/j.mcpro.2021.100140.

[42] Jordan T. Ash et al. "Joint analysis of expression levels and histological images identifies genes associated with tissue morphology". In: *Nature Communications* 12.1 (Mar. 2021). DOI: 10.1038/s41467-021-21727-x. URL: https://doi.org/10.1038/s41467-021-21727-x.

[43] Liviu Badea and Emil Stanescu. "Identifying transcriptomic correlates of histology using deep learning". In: *PLOS ONE* 15.11 (Nov. 2020). Ed. by Yan Chai Hum, e0242858. DOI: 10.1371/journal.pone.0242858. URL: https://doi.org/10.1371/journal.pone.0242858.

[44] Mathias Uhlen, Anita Bandrowski, Steven Carr, et al. "A proposal for validation of antibodies". In: *Nature Methods* 13.10 (Sept. 2016), pp. 823–827. DOI: 10.1038/nmeth.3995. URL: https://doi.org/10.1038/nmeth.3995.

[45] A. Schumacher, M. B. Rookmaaker, J. A. Joles, et al. "Defining the variety of cell types in developing and adult human kidneys by single-cell RNA sequencing". In: *npj Regenerative Medicine* 6.1 (Aug. 2021). DOI: 10.1038/s41536-021-00156-w. URL: https://doi.org/10.1038/s41536-021-00156-w.

[46] Ting Chen et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 1597–1607. URL: http://proceedings.mlr.press/v119/chen20j.html.

[47] Ting Chen, Simon Kornblith, Kevin Swersky, et al. "Big Self-Supervised Models are Strong Semi-Supervised Learners". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al. 2020. URL: https://proceedings.neurips.cc/paper/2020/hash/fcbc95ccdd551da181207c0c1400c655-Abstract.html.

[48] Daisuke Komura and Shumpei Ishikawa. "Machine Learning Methods for Histopathological Image Analysis". In: *Computational and Structural Biotechnology Journal* 16 (2018), pp. 34–42. DOI: 10.1016/j.csbj.2018.01.001. URL: https://doi.org/10.1016/j.csbj.2018.01.001.

[49] Sonal Kothari et al. "Removing Batch Effects From Histopathological Images for Enhanced Cancer Diagnosis". In: *IEEE Journal of Biomedical and Health Informatics* 18.3 (May 2014), pp. 765–772. DOI: 10.1109/jbhi.2013.2276766. URL: https://doi.org/10.1109/jbhi.2013.2276766.

[50] Gao Huang et al. "Densely Connected Convolutional Networks". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 2261–2269. DOI: 10.1109/CVPR.2017.243. URL: https://doi.org/10.1109/CVPR.2017.243.

[51] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2009. DOI: 10.1109/cvpr.2009.5206848. URL: https://doi.org/10.1109/cvpr.2009.5206848.

[52] Adam Paszke, Sam Gross, Francisco Massa, et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach et al. 2019, pp. 8024–8035. URL: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

[53] Edgar Riba, Dmytro Mishkin, Daniel Ponsa, et al. "Kornia: an Open Source Differentiable Computer Vision Library for PyTorch". In: *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*. IEEE, 2020, pp. 3663–3672. DOI: 10.1109/WACV45572.2020.9093363. URL: https://doi.org/10.1109/WACV45572.2020.9093363.

[54] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[55] Chan-Zuckerberg Biohub. *cellxgene Data Portal*. http://cellxgene.cziscience.com/collections/9b02383 9358-4f0f-9795-a891ec523bcc. Jan. 2022.

[56] Terrence F Meehan, Anna Maria Masci, Amina Abdulla, et al. "Logical Development of the Cell Ontology". In: *BMC Bioinformatics* 12.1 (Jan. 2011). DOI: 10.1186/1471-2105-12-6. URL: https://doi.org/10.1186/1471-2105-12-6.

[57] Masato Habuka, Linn Fagerberg, Björn M. Hallström, et al. "The Kidney Transcriptome and Proteome Defined by Transcriptomics and Antibody-Based Profiling". In: *PLoS ONE* 9.12 (Dec. 2014). Ed. by Harald Mischak, e116125. DOI: 10.1371/journal.pone.0116125. URL: https://doi.org/10.1371/journal.pone.0116125.

[58] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome Biology* 19.1 (Feb. 2018). DOI: 10.1186/s13059-017-1382-0. URL: https://doi.org/10.1186/s13059-017-1382-0.

[59] Malte D Luecken and Fabian J Theis. "Current best practices in single-cell RNA-seq analysis: a tutorial". In: *Molecular Systems Biology* 15.6 (June 2019). DOI: 10.15252/msb.20188746. URL: https://doi.org/10.15252/msb.20188746.

[60] Leland McInnes and John Healy. "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction". In: *CoRR* abs/1802.03426 (2018). arXiv: 1802.03426. URL: http://arxiv.org/abs/1802.03426.

[61] Fabian Pedregosa et al. "Scikit-Learn: Machine Learning in Python". In: *J. Mach. Learn. Res.* 12.null (Nov. 2011), pp. 2825–2830. ISSN: 1532-4435.

[62] Peter J. Thul et al. "A subcellular map of the human proteome". In: *Science* 356.6340 (May 2017). DOI: 10.1126/science.aal3321. URL: https://doi.org/10.1126/science.aal3321.

[63] A. Richard Kitching and Holly L. Hutton. "The Players: Cells Involved in Glomerular Disease". In: *Clinical Journal of the American Society of Nephrology* 11.9 (Apr. 2016), pp. 1664–1674. DOI: 10.2215/cjn.13791215. URL: https://doi.org/10.2215/cjn.13791215.

[64] T B Usdin et al. "Distribution of parathyroid hormone-2 receptor messenger ribonucleic acid in rat." In: *Endocrinology* 137.10 (Oct. 1996), pp. 4285–4297. DOI: 10.1210/endo.137.10.8828488. URL: https://doi.org/10.1210/endo.137.10.8828488.

[65] Alexander So and Bernard Thorens. "Uric acid transport and disease". In: *Journal of Clinical Investigation* 120.6 (June 2010), pp. 1791–1799. DOI: 10.1172/jci42344. URL: https://doi.org/10.1172/jci42344.

[66] Andrew Tao, Karan Sapra, and Bryan Catanzaro. "Hierarchical Multi-Scale Attention for Semantic Segmentation". In: *CoRR* abs/2005.10821 (2020). arXiv: 2005.10821. URL: https://arxiv.org/abs/2005.10821.

[67] Jingtao Guo et al. "The adult human testis transcriptional cell atlas". In: *Cell Research* 28.12 (Oct. 2018), pp. 1141–1157. DOI: 10.1038/s41422-018-0099-2. URL: https://doi.org/10.1038/s41422-018-0099-2.

[68] R.I. McLachlan et al. "Histological evaluation of the human testis – approaches to optimizing the clinical value of the assessment: Mini Review". In: *Human Reproduction* 22.1 (Aug. 2006), pp. 2–16. DOI: 10.1093/humrep/del279. URL: https://doi.org/10.1093/humrep/del279.

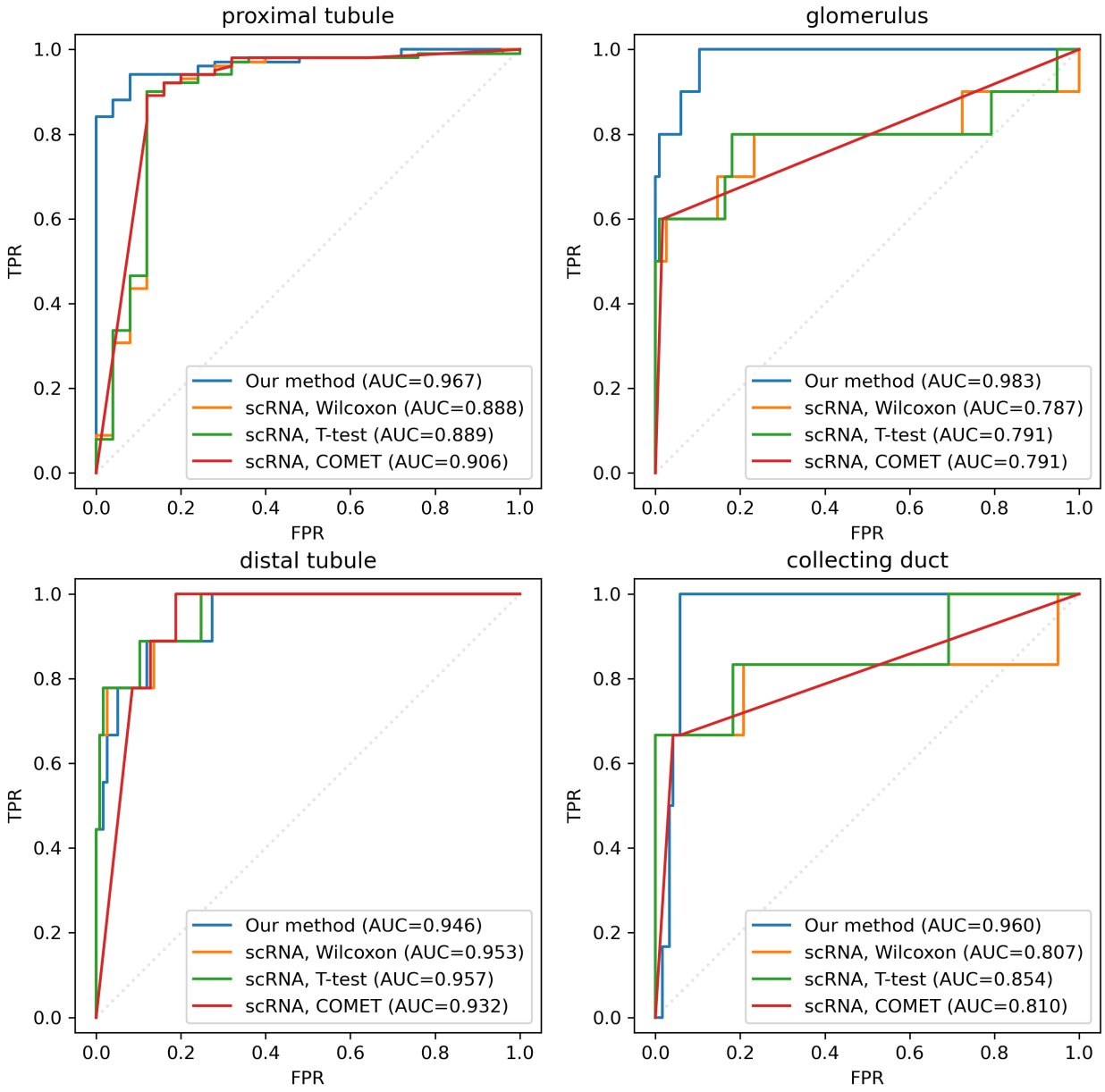

Figure 2.6: One-versus-rest ROC curves with respect to the labels of regional specificity in Habuka, Fagerberg, Hallström, et al. [57] for our classifier (blue) and the transcriptomic baselines.

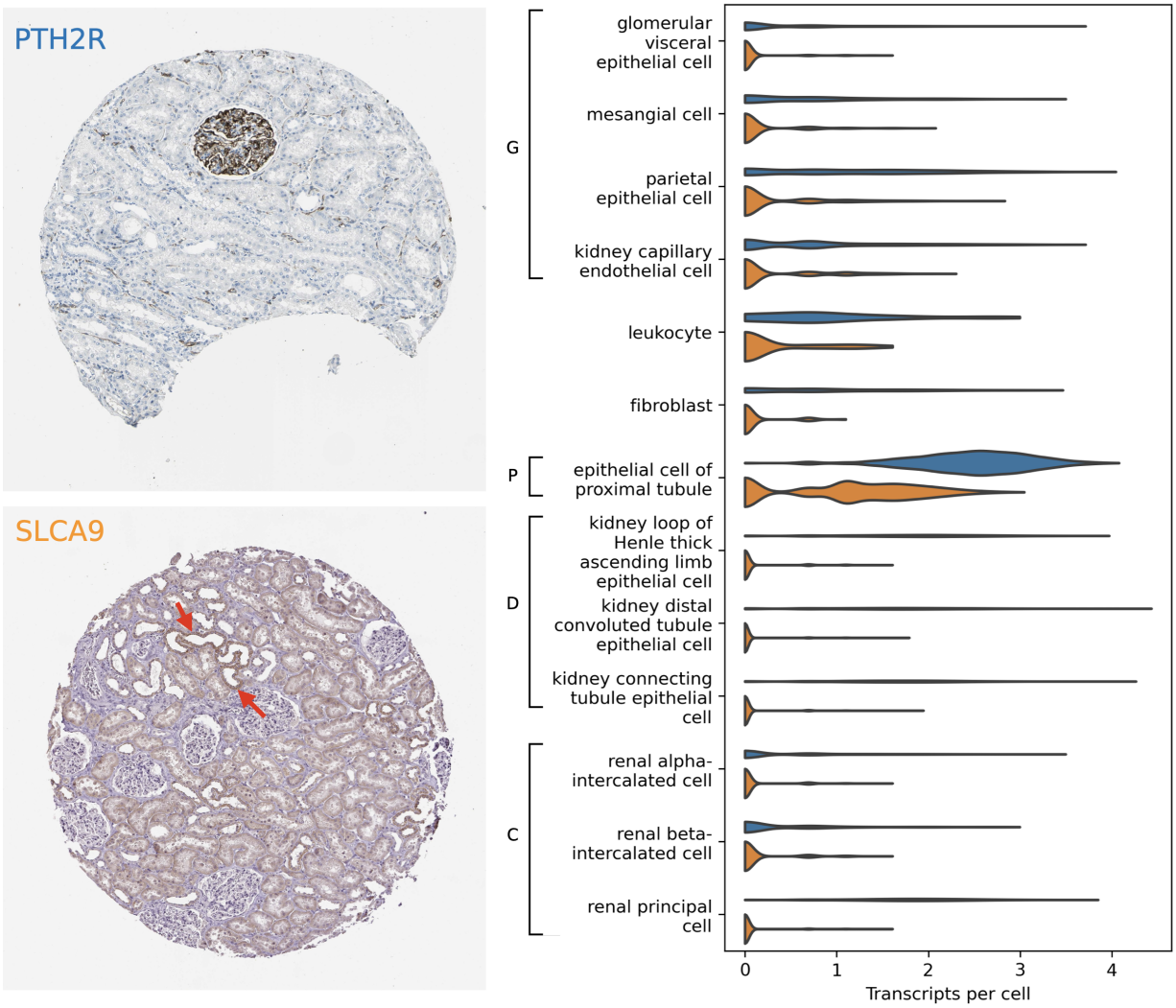

Figure 2.7: Example images of PTH2R (top; antibody HPA010655) and SLC2A9 (bottom; antibody HPA066229). Regions in the latter that appear to be stained distal tubules are indicated by red arrows. Violin plot indicates the transcriptional expression of PTH2R (blue) and SLC2A9 (orange) in each cell type of Muto et al. [19]. Brackets indicate anatomical regions (G: glomerulus; P: proximal tubule; D: distal tubule; C: collecting duct).

Chapter 3

Efficiently predicting high resolution mass spectra with graph neural networks

Authors: Michael Murphy, Stefanie Jegelka, Ernest Fraenkel, Tobias Kind, David Healey, Thomas Butler

This work was presented at ICML 2023 and published in the Proceedings of Machine Learning Research. I conceived the idea, developed the algorithms, carried out the analysis, and wrote the manuscript. Thomas Butler provided guidance on algorithmic directions and evaluations, and contributed to the introduction section. All authors provided commentary on the manuscript and input throughout.

3.1 Abstract

Identifying a small molecule from its mass spectrum is the primary open problem in computational metabolomics. This is typically cast as information retrieval: an unknown spectrum is matched against spectra predicted computationally from a large database of chemical structures. However, current approaches to spectrum prediction model the output space in ways that force a tradeoff between capturing high resolution mass information and tractable learning. We resolve this tradeoff by casting spectrum prediction as a mapping from an input molecular graph to a probability distribution over chemical formulas. We further discover that a large corpus of mass spectra can be closely approximated using a fixed vocabulary constituting only 2% of all observed formulas. This enables efficient spectrum prediction using an architecture similar to graph classification – GRAFF-MS – achieving significantly lower prediction error and greater retrieval accuracy than previous approaches.

3.2 Introduction

The identification of unknown small molecules in complex mixtures is a primary challenge in many areas of chemical and biological science. The standard high-throughput approach to small molecule identification is *tandem mass spectrometry* (MS/MS), with diverse appli-

cations including metabolomics [1], drug discovery [2], clinical diagnostics [3], forensics [4], and environmental monitoring [5].

MS/MS generates an experimental signature – a *mass spectrum* – of an unknown molecule by breaking it into fragments. The spectrum contains a (mass-to-charge ratio, height) tuple for each resulting fragment, reflecting its elemental composition, electric charge, and tendency to form. The problem of inferring the 2D structure of a molecule from its spectrum is known as *structural elucidation*. Structural elucidation is the primary computational bottleneck in MS/MS, and is far from solved: typically only 2−4% of spectra are identified in untargeted metabolomics experiments [6]. A recent competition – the 2022 Critical Analysis of Small Molecule Identification (CASMI) challenge [7] – saw no more than 30% accuracy, with algorithmic approaches only marginally outperforming manual annotation by expert chemists.

Because MS/MS is a lossy measurement, and available training sets are small, direct prediction of structures from spectra is particularly challenging. The approach for small molecule identification preferred in practice by most users of mass spectrometry is *spectral library search*, which casts the problem as information retrieval [8]: an observed spectrum is queried against a library of spectra with known structures. This provides an informative prior, and has the advantage of easy interpretability. But as there are relatively few (10^4) small molecules with publicly known experimental mass spectra, in spectral library search it is necessary to augment libraries with spectra predicted from large databases ($10^6 - 10^9$) of molecular graphs. This motivates the problem of *spectrum prediction*: the ability to predict higher-quality spectra from large chemical structure databases could greatly increase compound identification rates in real experimental settings.

Spectrum prediction is actively studied in metabolomics and quantum chemistry [9], yet has historically received little attention from the machine learning community. A major challenge in spectrum prediction is modelling the output space: a mass spectrum is a variable-length set of real-valued tuples, which is not straightforward to represent as an output of a machine learning model. The mass-to-charge (m/z) coordinate poses particular difficulty: it must be predicted with high precision, as a key strength of MS/MS is the ability to distinguish small fractional m/z differences (on the order of 10^{-6}) representative of different elemental composition.

Previous approaches to spectrum prediction force a tradeoff between capturing high resolution m/z information and tractability of the learning problem. *Mass-binning* methods [10, 11, 12] represent a spectrum as a fixed-length vector by discretizing the m/z axis at regular intervals, discarding fine-scale information in favor of tractable learning. *Bond-breaking* methods [13, 14] achieve perfect m/z resolution, but use expensive combinatorial enumeration of substructures.

Our work presents a novel formulation that exploits the many-to-one relationship between molecular graphs and chemical formulas. Specifically, we make the following contributions:

- We formulate spectrum prediction as a mapping from a molecular graph to a probability distribution over chemical formulas, allowing full resolution predictions without enumerating substructures;
- We discover most mass spectra can be effectively approximated with a small fixed vocabulary of chemical formulas, bypassing the tradeoff between m/z resolution and

tractable learning; and

- We implement an efficient graph neural network architecture, GRAFF-MS, that outperforms state-of-the-art in both prediction error and runtime on canonical mass spectrometry datasets, and yields superior accuracy on a large-scale structure retrieval task.

3.3 Background

We denote vectors $\boldsymbol{x}$ in bold lowercase and matrices $\boldsymbol{X}$ in bold uppercase.

A *molecular graph* $G = (V, E, \boldsymbol{a}, \boldsymbol{b})$ is a minimal description of the 2D structure of a molecule: it comprises an undirected graph with nodes V representing atoms, and edges $E \subset V \times V$ representing bonds. Each node i is labelled with a chemical element $a_i \in \{\text{C, H, N, O, P, S} \dots\}$, and each edge (i, j) with a bond order $b_{ij} \in \{1, 1.5, 2, 3\}$.

A *chemical formula* f (e.g. $C_8H_{10}N_4O_2$) describes a multiset of atoms, which we encode as a nonnegative integer vector of atom counts in $\mathcal{F}^* \doteq \mathbb{Z}_+^{\{\text{C, H, N, O, P, S} \dots\}}$. Formulas may be added and subtracted from one another, and inequalities between formulas are taken to hold elementwise. The *subformulas* of f are the set $\mathcal{F}(f) \doteq \{f' \in \mathcal{F}^* : f' \leq f\}$.

$\langle \mu, f \rangle \in \mathbb{R}_+$ is the *theoretical mass* of formula f, in units of daltons (Da): this is a weighted sum of the monoisotopic masses of the elements of the periodic table, $\mu \in \mathbb{R}_+^{\{\text{C, H, N, O, P, S} \dots\}}$, with multiplicities given by f.

A *mass spectrum* S is a variable-length set of *peaks*, each of which is a (m/z, intensity) tuple $(m_i, y_i) \in \mathbb{R}_+^2$. We use the notation $i \in S$ to index peaks in a spectrum. We assume spectra are normalized, permitting us to treat them as probability distributions: $\sum_{i \in S} y_i = 1$. A mass spectrum is implicitly always accompanied by a precursor formula P. We always assume charge $z = 1$, as it is rare for small molecules to acquire more than a single charge.

3.3.1 Tandem mass spectrometry

A tandem mass spectrometer is a scientific instrument that generates high-throughput experimental signatures of the molecules present in a complex mixture. It operates by ionizing a chemical sample into a jet of electrically-charged gas. This gas is electromagnetically filtered to select a population of *precursor ions* of a specific mass-to-charge ratio (m/z) representing a unique molecular structure. Each precursor ion is fragmented by collision with molecules of an inert gas. If a collision occurs with sufficient energy, one or more bonds in the precursor will break, yielding a charged *product ion* and one or more uncharged *neutral loss* molecules. The product ion is measured by a detector, which records its m/z up to a small error proportional to m/z times the instrument resolution ϵ, typically on the order of 10^{-6}. This process is repeated for large numbers of identical precursor ions, building up a histogram indexed by m/z. Local maxima in this histogram are termed *peaks*: ideally, each peak represents a unique product ion, with intensity reflecting its probability of formation. This set of peaks constitutes the mass spectrum. A typical mass spectrometry experiment acquires mass spectra for tens of thousands of distinct precursors in this manner. We depict

this process in Figure 3.1A, and the relationship between precursor ion, product ion and neutral loss in Figure 3.1B.

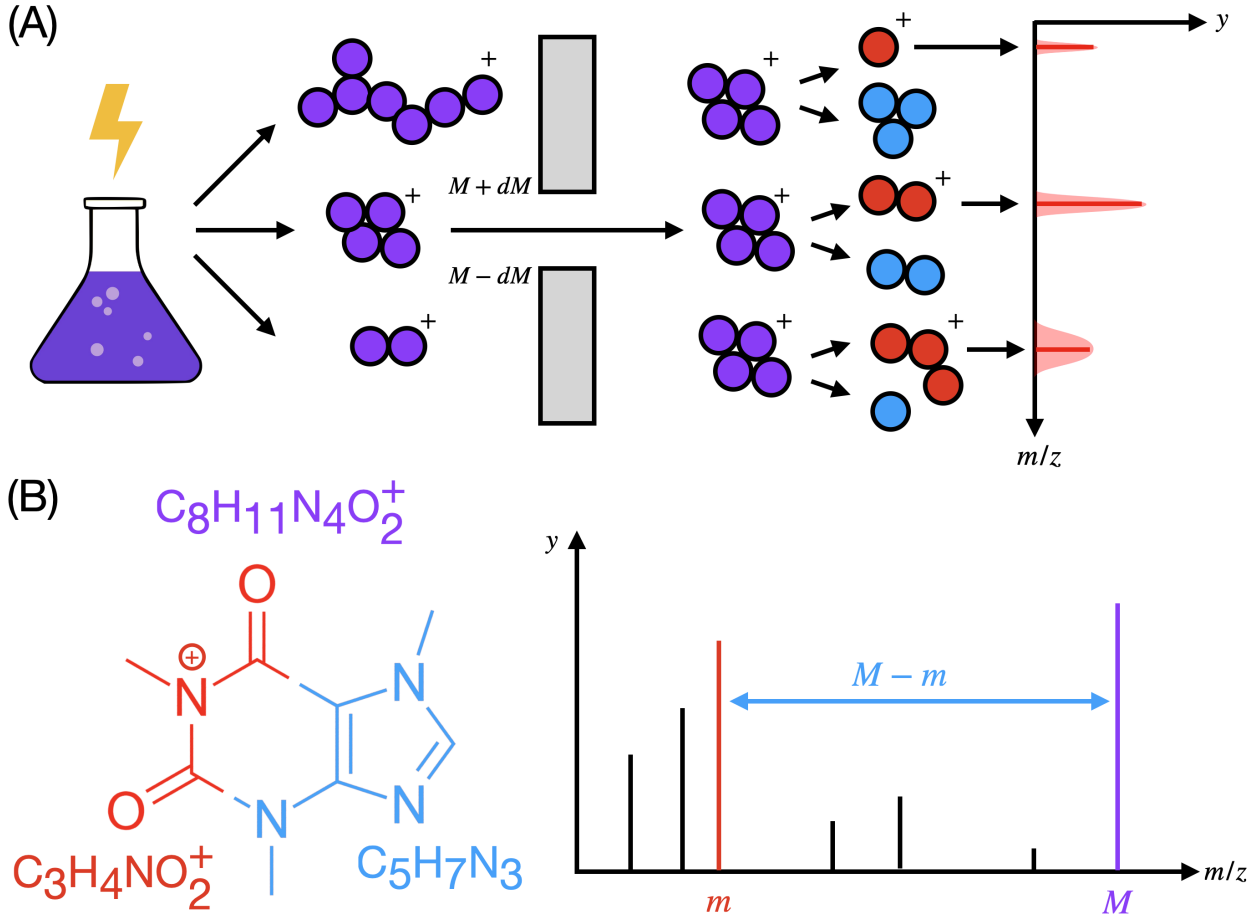

Figure 3.1: (A) The workflow of tandem mass spectrometry. A chemical mixture is ionized and filtered to isolate precursor ions of $m/z = M$; these are fragmented into product ions (red) and neutral losses (blue), and a detector yields a histogram of product ions indexed by m/z, with measurement error proportional to m/z. (B) An example fragmentation of a precursor ion with formula $C_8H_{11}N_4O_2$. Fragmentation breaks bonds, cutting the molecular graph into connected components. The component retaining the charge is the product ion; its complement is the neutral loss. The peak at $m/z = m$ represents the product ion; given the precursor formula, we can equally specify this peak by its formula $C_3H_4NO_2$, or the formula of its neutral loss $C_5H_7N_3$.

3.3.2 Mass decomposition

Modern mass spectrometry achieves sufficiently high resolution to detect small deviations in m/z from integrality that are characteristic of different chemical elements. This property is a key strength of the technology, because it permits annotating peaks with formulas

through *mass decomposition* [15]. Given a product ion of $m/z = m$, a precursor formula P, and an instrument resolution ϵ, *product mass decomposition* yields a set $\mathcal{F}(P, m, \epsilon)$ of chemically plausible subformulas of P whose theoretical masses lie within the (multiplicative) measurement error ϵm of m. This can be cast as the following integer program, in which all solutions with cost $\leq \epsilon m$ are enumerated:

$$\min_{f \in \mathcal{F}^*} |\langle \mu, f \rangle - m| \tag{3.1}$$

$$\text{s.t. } f \leq P, f \in \Omega \tag{3.2}$$

where Ω describes a general set of constraints that exclude unrealistic chemical formulas [16]. For modern instruments, ϵ is sufficiently small for there to typically be only one or a few valid solutions. This allows us to later rely on product mass decomposition as a black-box to generate useful formula annotations at training time.

3.4 Related work

Bond-breaking is the most studied approach to spectrum prediction [17, 13, 14, 18]. This solves the problem of representing the output space by enumerating the structures of all probable product ions: these are taken to be connected subgraphs of the precursor, generated by sequences of edge removals. Each product ion structure is scored for its probability of formation, and a spectrum is generated by associating this probability with each structure's theoretical m/z. Bond-breaking therefore achieves perfect m/z resolution, but suffers from two major weaknesses: first, enumerating substructures scales poorly with molecule size, and is not conducive to massively-parallel implementation on a GPU. We found a state-of-the-art method [13] takes ∼5s on average to predict a single mass spectrum, which precludes training on the largest available datasets: using the same settings as its authors, training [13] on ∼300k spectra in NIST-20 would take an estimated *three months* on a 64-core machine. It also poses serious limitations at test time, as inference with a large-scale structure database like ChEMBL [19] requires predicting millions of spectra. The other weakness of bond-breaking arises from a restrictive modelling assumption: *rearrangement reactions* [20] frequently yield product ions that are not reachable from the precursor by sequences of edge removals.[1]

Mass-binning is used for spectrum prediction by [10], and subsequently employed in recent preprints [11, 12]. This approach represents a mass spectrum as a fixed-length vector via discretization: the m/z axis is partitioned into narrow regularly-spaced bins, and each bin is assigned the sum of the intensities of all peaks falling within its endpoints. Spectrum prediction then becomes a vector-valued regression problem, which is conducive to GPU implementation and scales better than bond-breaking. But because a target space with millions of mass bins is too large, realistic bin counts lose essential high resolution information about the chemical formulas of the peaks: discarding a key strength of MS/MS analysis in favor of a tractable learning problem. Such models are also susceptible to edge effects, where

[1] While engineered rules are used in bond-breaking to account for certain well-studied rearrangements, we found the state-of-the-art method CFM-ID still fails to assign a formula annotation within 10ppm to 42% of monoisotopic peaks in the NIST-20 dataset.

m/z measurement error of the instrument can cause peaks of the same product ion to cross bin boundaries from spectrum to spectrum.

Other approaches include molecular dynamics simulation [21], which has extremely high computational costs; and NLP-inspired models for peptides [22, 23], which are effective but inapplicable to other types of molecules.

While this manuscript was under review, two GNN-based approaches to modelling mass spectra as distributions over subformulas were also published. Rather than the fixed vocabulary approximation we discover here, Zhu and Jonas [24] use an exhaustive enumeration scheme to generate subformulas, which are then used in a formula-to-atom attention operation to predict a peak height for each. Goldman, Li, and Coley [25] decode a prefix-tree of plausible subformulas from the molecular graph and predict intensities for these using a set-to-set transformer.

3.5 GrAFF-MS

Our approach comprises three major components: first, we represent the output space of spectrum prediction as a space of probability distributions over chemical formulas. We then introduce a constant-sized approximation of this output space using a fixed vocabulary of formulas, which we can generate from our training data; we later show this introduces only minor approximation cost, as most formulas occur with low probability. Finally, we derive a loss function that takes into account data-specific ambiguities introduced by our model of the output space. These components together allow us to efficiently predict spectra using a standard graph neural network architecture.

We call our approach GrAFF-MS: (Gr)aph neural network for (A)pproximation via (F)ixed (F)ormulas of (M)ass (S)pectra.

3.5.1 Modelling spectra as probability distributions over chemical subformulas of the precursor

Our aim is to predict a mass spectrum from a molecular graph. To do so, we must determine how to best represent the output space: a spectrum consists of a variable-length set of peaks located at continuous m/z positions, whose heights sum to one. We notice that peaks are not located arbitrarily: the set of m/zs is structured, as the m/z of a peak is determined (up to measurement error) by the chemical formula of its corresponding product ion. This formula is sufficient to determine the m/z; in particular, we do not need to know the product ion's full 2D structure. We therefore model a mass spectrum as a probability distribution over *chemical subformulas* $\mathcal{F}(P)$ of the precursor P:

$$S = \{(m_f, y_f) : f \in \mathcal{F}(P)\} \tag{3.3}$$

where $m_f \doteq \langle \mu, f \rangle$ is the theoretical mass of formula f.

This is more efficient in principle than bond-breaking, which models a spectrum as a distribution over at worst exponentially many *substructures* of the precursor. In contrast, the number of *subformulas* is only polynomial in the coefficients of the precursor formula –

and the majority of subformulas can be ruled out *a priori* as chemically infeasible [16]. It is also less restrictive than bond-breaking, which relies on hand-engineered rules to capture rearrangement reactions: enumerating subformulas is guaranteed to cover all possible peaks, irrespective of whether the structure of their product ion is reachable by edge removals or not. Yet our approach preserves the core advantage of bond-breaking over mass-binning: predicting a height for each subformula yields spectra with perfect m/z resolution.

3.5.2 Fixed vocabulary approximation of formula space

In practice, enumerating subformulas is still a costly operation for larger molecules. One way to avoid this would be to sequentially decode formulas of nonzero probability one at a time: we opt not to do so, as this requires a more complex, data-hungrier model, and necessitates a linear ordering of formulas, for which there is not an obvious correct choice. Instead, we exploit a property of small molecule mass spectra that we discovered in this work and illustrate in Figure 3.2: almost all of the signal in small molecule mass spectra lies in peaks that can be explained by a relatively small number ($\sim$2%) of product ion and neutral loss formulas that frequently recur across spectra.

Inspired by this finding, we approximate $\mathcal{F}(P)$ via the union $\hat{\mathcal{F}}(P) = \hat{\mathcal{P}} \cup (P - \hat{\mathcal{L}})$ of a fixed set of frequent product ion formulas $\hat{\mathcal{P}}$ and a variable set of 'precursor-specific' formulas $P - \hat{\mathcal{L}}$ obtained by subtracting a fixed set of frequent neutral loss formulas $\hat{\mathcal{L}}$ from the precursor P. This greatly simplifies the spectrum prediction problem: we now only need to predict a probability for each of the formulas in $\hat{\mathcal{P}}$ and $\hat{\mathcal{L}}$, which we can accomplish with time *constant* in the size of the precursor.

Stated explicitly, we approximate the spectrum as:

$$S \approx \{(m_f, y_f) : f \in \hat{\mathcal{F}}(P)\} \tag{3.4}$$

where a height of zero is implicitly assigned to any formula not in $\hat{\mathcal{F}}(P)$.

The fact that we can equally represent a product ion by either its own formula or a neutral loss formula relative to its precursor ion is crucial to generalization, as also noted by Wei et al. [10]. If we only included frequent product ion formulas, we would explain peaks of low mass well, which typically correspond to small charged functional groups. But as formula space becomes larger with increasing mass, it becomes increasingly unlikely that every significant peak of higher mass in an unseen compound will be explained. However, such peaks do not represent arbitrary subformulas of the precursor: they tend to arise from losses of small uncharged functional groups and combinations thereof, which we capture by including frequent neutral losses.

Our algorithm to generate $\hat{\mathcal{P}}$ and $\hat{\mathcal{L}}$ involves listing all product ion and neutral loss formulas yielded by mass decomposition of the training set, and ranking them by the sum of the heights of all peaks to which each formula is assigned; we select the top K highest ranked among either type. The algorithm is provided in appendix 3.9.1.

3.5.3 Peak-marginal cross entropy

To train our approach, we must rely on formula annotations generated by mass decomposition. Because mass spectrometers have limited resolution, often more than one valid

subformula has a mass within measurement error of a peak. These are considered equiprobable *a priori*, and need not be mutually exclusive: it is possible for a compound to contain two distinct substructures with m/z difference smaller than the measurement error. As we cannot pick a single formula in such cases, we approximate the full cross entropy loss by marginalizing over compatible formulas: we term this the *peak-marginal cross entropy*. We minimize this loss with respect to the parameters of a neural network $\hat{\boldsymbol{y}}(\cdot;\theta)$:

$$\min_{\theta} \; -\sum_{n=1}^{N} \sum_{i \in S_n} y_i^n \log \sum_{f \in \hat{\mathcal{F}}_i^n} \hat{y}_f(G_n;\theta) \tag{3.5}$$

using $\hat{\mathcal{F}}_i^n \doteq \hat{\mathcal{F}}(P_n) \cap \mathcal{F}(P_n, m_i^n, \epsilon)$ to indicate the intersection of our fixed vocabulary with the formula annotations for peak i of spectrum n. We provide a derivation from first principles in appendix 3.9.2.

In this formulation, given a molecular graph G of a precursor with formula P, our model predicts a probability $\hat{y}_f$ for every formula f in the fixed vocabulary. This produces a spectrum $\hat{S} = \{(m_f, \hat{y}_f) : f \in \hat{\mathcal{F}}(P)\}$. These per-formula probabilities are summed within each observed peak across its compatible formulas to yield a predicted peak height, and the cross-entropy between the observed and predicted peak heights across the entire spectrum is minimized.

3.5.4 Model architecture

Formulating spectrum prediction as graph classification permits applying a typical GNN architecture. GRAFF-MS uses a graph isomorphism network with edge and graph Laplacian features [26, 27, 28]. This encodes the molecular graph by a dense vector representation, which is then conditioned on mass-spectral covariates and passed through a feed-forward network that decodes a logit for each formula in the vocabulary.

We start with the graph of the 2D structure $G = (V, E)$, to which we add a virtual node [29] and four classes of features: node features $\boldsymbol{a}_i \in \mathbb{R}^{d_{atom}}$, edge features $\boldsymbol{b}_{ij} \in \mathbb{R}^{d_{bond}}$, covariate features $\boldsymbol{c} \in \mathbb{R}^{d_{cov}}$, and the top eigenvectors and eigenvalues of the graph Laplacian $\boldsymbol{v}_i \in \mathbb{R}^{d_{eig}}, \boldsymbol{\lambda} \in \mathbb{R}^{d_{eig}}$. We use the canonical atom and bond featurizers from `DGL-LifeSci` [30] to generate $\boldsymbol{a}$ and $\boldsymbol{b}$. Since a mass spectrum is not fully determined by the molecular graph, $\boldsymbol{c}$ includes a number of necessary experimental parameters: normalized collision energy, precursor ion type, instrument model, and presence of isotopic peaks. Further details are provided in Table 3.3 in the appendix; there we also provide hyperparameter settings.

We first embed the node, edge, and covariate features into $\mathbb{R}^{d_{enc}}$, reusing the following `MLP` block:

$$\texttt{MLP}(\cdot) = \texttt{LayerNorm}(\texttt{Dropout}(\texttt{SiLU}(\texttt{Linear}(\cdot))))$$

and transform the Laplacian features into node positional encodings in $\mathbb{R}^{d_{enc}}$ using a SignNet [28] with ϕ and ρ both implemented as 2-layer stacked `MLP`s:

$$\boldsymbol{x}_i^{atom} = \texttt{MLP}_{atom}(\boldsymbol{a}_i) \tag{3.6}$$

$$\boldsymbol{x}_i^{eig} = \texttt{SignNet}(\boldsymbol{v}_i, \boldsymbol{\lambda}) \tag{3.7}$$

$$\boldsymbol{x}_{ij}^{bond} = \texttt{MLP}_{bond}(\boldsymbol{b}_{ij}) \tag{3.8}$$
$$\boldsymbol{x}^{cov} = \texttt{MLP}_{cov}(\boldsymbol{c}) \tag{3.9}$$

taking $i \in V$ and $(i, j) \in E$. We sum the embedded atom features and node positional encodings, and pass these along with embedded bond features into a stack of alternating L message-passing layers to update the node representations, and L `MLP` layers to update the edge representations.

$$\boldsymbol{x}_i^{(0)} = \boldsymbol{x}_i^{atom} + \boldsymbol{x}_i^{eig} \tag{3.10}$$
$$\boldsymbol{e}_{ij}^{(0)} = \boldsymbol{x}_{ij}^{bond} \tag{3.11}$$
$$\boldsymbol{X}^{(l+1)} = \boldsymbol{X}^{(l)} + \texttt{GINEConv}^{(l)}(G, \boldsymbol{X}^{(l)}, \boldsymbol{E}^{(l)}) \tag{3.12}$$
$$\boldsymbol{e}_{ij}^{(l+1)} = \boldsymbol{e}_{ij}^{(l)} + \texttt{MLP}_{edge}^{(l)}(\boldsymbol{e}_{ij}^{(l)} \| \boldsymbol{x}_i^{(l+1)} \| \boldsymbol{x}_j^{(l+1)}) \tag{3.13}$$

where $\|$ denotes concatenation. The message-passing layer uses the `GINEConv` operation implemented in [31]: for its internal feed-forward network, we use two stacked `MLP` blocks with GraphNorm [32] in place of layer normalization. We similarly replace layer normalization with GraphNorm in the $\texttt{MLP}_{edge}$ blocks. Both node and edge updates use residual connections, which we found greatly accelerate training.

We generate a dense representation of the molecule by attention pooling over nodes [33], to which we add the embedded covariate features. An MLP decodes this into a spectrum representation $\boldsymbol{x}^{spec} \in \mathbb{R}^{d_{dec}}$:

$$a_i = \texttt{Softmax}_{i \in V}(\texttt{Linear}(\boldsymbol{x}_i)) \tag{3.14}$$
$$\boldsymbol{x}^{mol} = \sum\nolimits_{i \in V} a_i \boldsymbol{x}_i^{(L)} \tag{3.15}$$
$$\boldsymbol{x}^{spec} = \texttt{MLP}_{spec}(\boldsymbol{x}^{mol} + \boldsymbol{x}^{cov}), \tag{3.16}$$

where $\texttt{MLP}_{spec}$ is a stack of L' `MLP` blocks with residual connections. In principle, we may now project this representation via a linear layer $(\boldsymbol{w}_k, b_k)$ into a logit z_k for each of the K product ion or neutral loss formulas in the vocabulary.

3.5.5 Domain-specific modifications

We must now introduce some corrections motivated by domain knowledge to produce realistic mass spectra.

Depending on instrument parameters, tandem mass spectra can display small peaks arising from higher isotopic states of the precursor ion, at integral m/z shifts relative to the monoisotopic peak. We model this as a source of noise: rather than expanding our vocabulary, we apply to all predictions a scalar offset for each isotopic state $\delta \in \{0, 1, 2\}$, which we parameterize as a linear function of $\boldsymbol{x}^{spec}$.

As our vocabulary includes both product ions and neutral losses, we also face occasional *double-counting*: depending on the precursor P, there are cases where the same subformula f will be predicted both as a product ion ($f \in \hat{\mathcal{P}}$) and a neutral loss ($P - f \in \hat{\mathcal{L}}$). In such cases we subtract a $\log(2)$ correction factor from both logits: this way the innermost summation in Equation (3.5) takes the average of their contributions instead of their sum.

Applying these corrections and softmaxing yields the final heights of the predicted mass spectrum $\hat{\boldsymbol{y}}$:

$$z_{k\delta} = \boldsymbol{w}_k \boldsymbol{x}^{spec} + \boldsymbol{w}_\delta \boldsymbol{x}^{spec} + b_k - \mathbb{I}[k \text{ is double-counted}] \log(2) \tag{3.17}$$

$$\hat{y}_{k\delta} = \texttt{Softmax}_{k\delta}(z_{k\delta}) \tag{3.18}$$

where the subscript k is an index into our fixed vocabulary.

3.6 Experiments

3.6.1 Datasets

NIST-20

We train our model on the NIST-20 tandem MS library [34]. This is the largest commercial dataset of high resolution mass spectra of small molecules, curated by expert chemists, and is available for a modest fee.[2] For each measured compound, NIST-20 provides typically several spectra acquired across a range of collision energies. Each spectrum is represented as a list of (m/z, intensity, annotation) peak tuples, in addition to metadata describing instrumental parameters and compound identity. The annotation field includes a list of formula hypotheses per peak that were computed by NIST using mass decomposition and verified by expert chemists.

We restrict NIST-20 to HCD Orbitrap spectra with $[M+H]^+$ or $[M-H]^-$ precursor ions. We exclude structures that are annotated as glycans or peptides or exceed 1000Da in mass (as these are not typically considered small molecules) or have atoms other than {C, H, N, O, P, S, F, Cl, Br, I}.

We use an 80/10/10 structure-disjoint train/validation/test split, which we generate by grouping spectra according to the connectivity substring of their InChIKey [36], and assigning groups of spectra to splits. As the baseline CFM-ID only predicts monoisotopic spectra at qualitative energy levels {low, medium, high}, we restrict the test set to spectra with corresponding energies $\{20, 35, 50\}$ in which no peaks were annotated as higher isotopes. This yields 287,995 (18,665) training, 36,265 (2,346) validation, and 4,424 (1,632) test spectra (structures).

CASMI-16

It is well known that uniform train-test splitting can overestimate generalization in molecular machine learning [37]. To address this issue, we employ an independent test set: the spectra of the 2016 CASMI challenge [38]. This is a small public-domain mass spectrometry

[2] Open data is not the norm in small molecule mass spectrometry, as large-scale annotated data has commercial value and requires substantial time commitment from teams of highly-trained human experts. No public-domain dataset comparable to NIST-20 therefore exists. However, NIST-20 and its predecessors are commonplace in academic mass spectrometry, and have been used in ML research [10, 35].

dataset, constructed by domain experts specifically for benchmarking algorithms, and comprises structures selected as representative of those encountered 'in the wild' when performing mass spectrometry of small molecules.

We use $[M+H]^+$ and $[M-H]^-$ spectra from the combined 'Training' and 'Challenge' splits from Categories 2 and 3 of the challenge. We exclude any structures from CASMI-16 with an InChIKey connectivity match to any in NIST-20, yielding 166 spectra of 151 structures. CASMI-16 spectra are acquired with collision energy stepping, which generates a mixed spectrum from energies of $\{20, 35, 50\}$; for all methods we approximate this by predicting only the middle energy.

GNPS

To simulate performance in a real experimental setting, we extracted a subset of spectra from GNPS [39] that represent natural product molecules not found in NIST-20. This is a challenging dataset, as GNPS spectra are contributed by the community in an uncurated manner, and often are missing key covariates for spectrum prediction. To exclude obvious poor-quality spectra, we only consider $[M+H]^+$ and $[M-H]^-$ Orbitrap spectra, with reported precursor m/z matching the theoretical mass. GNPS does not report collision energy; we assume energy = 35, and only include spectra with a (number of peaks) to (precursor m/z) ratio between the 10^{th} and 90^{th} percentiles of that quantity for NIST-20 spectra acquired at that energy. This results in 677 mass spectra of 606 structures.

3.6.2 Baselines

CFM-ID

CFM-ID [13] is a bond-breaking method, viewed by the mass spectrometry community as the state-of-the-art in spectrum prediction [9]. We found CFM-ID prohibitively expensive to train on NIST-20 (one parallelized EM iterate on a subset of ∼60k spectra took 10 hours on a 64-core machine) so we use trained weights provided by its authors, learned from 18,282 spectra in the commercial METLIN dataset [40]. Domain experts consider spectra acquired under METLIN's conditions interchangeable with those of NIST-20 [41] so it is reasonable to evaluate their model on our data.

NEIMS

NEIMS [10] is a feed-forward network that inputs a precomputed molecular fingerprint and outputs a mass-binned spectrum, which is postprocessed using a domain-specific gating operation ("bidirectional prediction"). As NEIMS was originally developed for electron-impact mass spectra, we retrained NEIMS on NIST-20, which necessitated two modifications: (1) we concatenate a vector of covariates to the fingerprint vector, without which NIST-20 spectra are not fully determined; and (2) we bin at 0.1Da intervals instead of 1Da intervals, to account for finer instrument resolution in NIST-20. We otherwise use the same hyperparameter settings as the original paper, and early-stop on validation loss.

3.6.3 Evaluation

Cosine similarity

We evaluate predictive accuracy against ground truth spectra via *mass-spectral cosine similarity* [42]. This modifies the standard cosine similarity to allow for inexact matches in the m/z coordinates between two spectra. For two spectra S and $\hat{S}$, mass-spectral cosine similarity $C_{S,\hat{S}}$ is the maximal cosine similarity between vectors of peak heights taken over all peak matchings within a mass tolerance window τ. It is computed by solving a linear sum assignment problem:

$$C_{S,\hat{S}} \doteq \max_{x_{ij}\in\{0,1\}} \sum_{\substack{i\in S,\ j\in\hat{S}:\\ |m_i-\hat{m}_j|\le\tau}} x_{ij} \frac{y_i}{\|y\|_2} \frac{\hat{y}_j}{\|\hat{y}\|_2} \tag{3.19}$$

$$\text{s.t.} \quad \textstyle\sum_{i\in S} x_{ij} \le 1 \tag{3.20}$$

$$\textstyle\sum_{j\in\hat{S}} x_{ij} \le 1 \tag{3.21}$$

We use the `CosineHungarian` implementation from `matchms` [43], with tolerance $\tau = 0.1$Da.

Time complexity

We also empirically compare dependence of runtime on input size between GRAFF-MS and the bond-breaking method CFM-ID. For fair comparison, we time a forward pass for each structure in the NIST-20 test split using only the CPU, without any batching. We include time spent in preprocessing: our input is a SMILES string and experimental covariates, and our output is a spectrum. As collision energy affects the number of fragments that CFM-ID generates, we predict spectra at low, medium, and high energies and use the average runtime of the three.

Spectral library search

We characterize retrieval performance on a large-scale spectral library search task. For each method, we predict a library of mass spectra from the structures in NIST-20, which we augment with 200k decoy structures sampled from ChEMBL within ±0.1Da of any NIST-20 structure. $[M+H]^+$ and $[M-H]^-$ spectra are predicted at collision energies $\{20, 35, 50\}$, resulting in a library of 1,262,025 spectra of 221,502 structures. We query each of the 4,424 experimental spectra from the NIST-20 test split against this library, restricting comparisons to spectra with the same ionization mode and collision energy, of theoretical m/z within 0.1Da of the query. We rank the resulting matches by mass-spectral cosine similarity, and compute recall-at-k: both of the correct 2D structure, and of any 2D structure with the correct chemical formula.

3.7 Results

3.7.1 A fixed vocabulary of products and losses captures the vast majority of fragmentation events

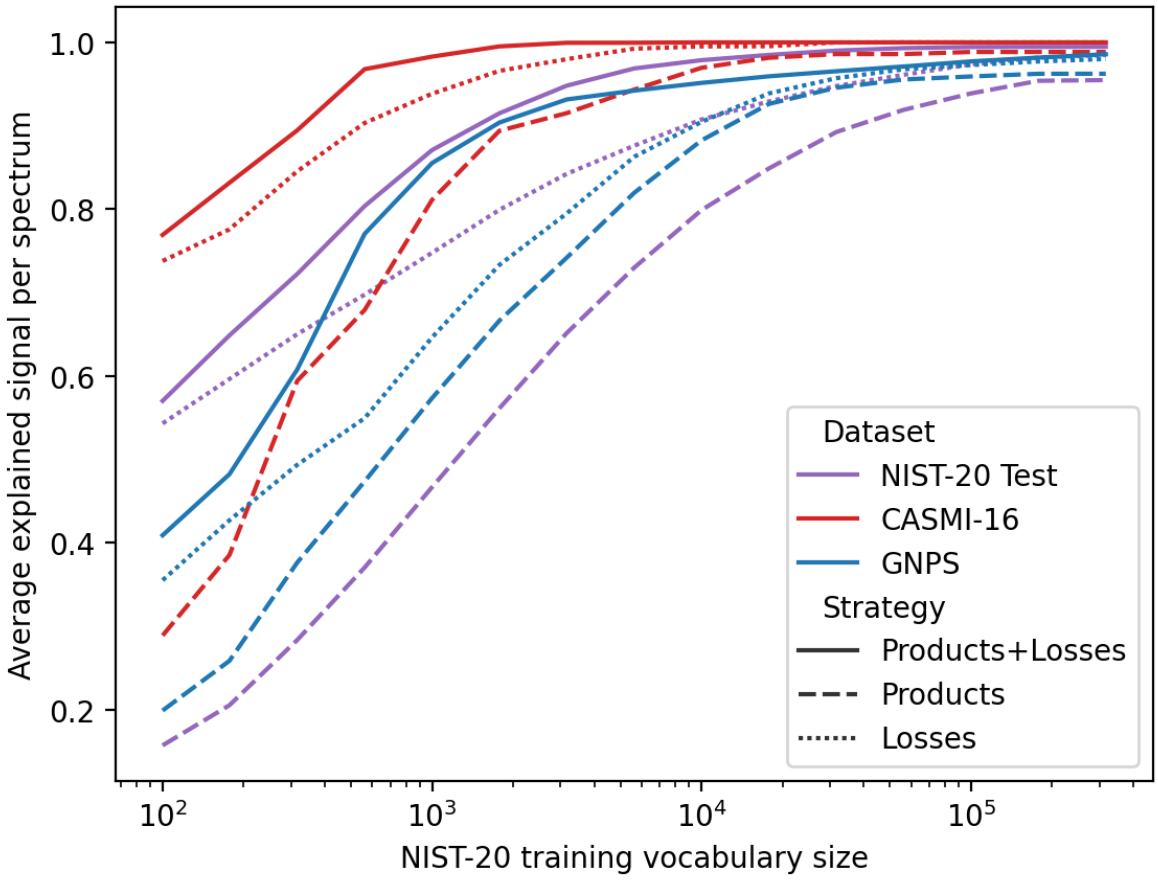

Figure 3.2: Generalization of different heuristics for fixed-size vocabulary selection. For a given vocabulary size on the x-axis, the y-axis indicates the sum of all explained peaks' heights within a given spectrum, averaged over all spectra.

Figure 3.2 illustrates the fraction of ion counts explained on average across mass spectra as the vocabulary size is varied. We observe most signal lies within peaks explainable by a relatively small number of product ion and neutral loss formulas. In particular, the vocabulary of $K = 10^4$ formulas we select from the NIST-20 training split (which contains 188,349 unique product ion and 351,165 unique neutral loss formulas) is sufficient to explain 98% of ion counts in the structure-disjoint test split. This vocabulary generalizes beyond NIST-20 to both CASMI-16 and GNPS, suggesting this 'formula sparsity' is a general property of small molecule mass spectra.

We also compare alternative strategies of picking only the top product ions or top neutral losses: our approach of using both types of formulas explains more signal for a fixed K than either type alone.

3.7.2 GRAFF-MS outperforms bond-breaking and mass-binning on standard MS/MS datasets

Table 3.1 shows GRAFF-MS produces spectra with greater cosine similarity to ground-truth than either baseline. These results hold for our test split of NIST-20 and for the independent test sets CASMI-16 and GNPS. We see all methods perform better on CASMI-16 than NIST-20: this is likely because NIST-20 includes a minority of substantially larger molecules (max weight 995Da) than CASMI-16 (max weight 539Da), with which all three methods struggle. Conversely, the poorer performance of all methods on GNPS likely reflects the difficulty of predicting the noisier spectra acquired in real-world biological experiments, as compared to curated spectra generated from libraries of pure chemical standards.

Table 3.1: Mean cosine similarity between predicted and true spectra on the NIST-20 test split, CASMI-16, and GNPS. 95% confidence intervals are computed via nonparametric bootstrap.

	NIST-20 TEST ($N = 4424$)	CASMI-16 ($N = 166$)	GNPS ($N = 677$)
CFM-ID	$0.53 \pm .01$	$0.71 \pm .04$	$0.26 \pm .02$
NEIMS	$0.60 \pm .01$	$0.57 \pm .06$	$0.29 \pm .02$
GRAFF-MS	$\mathbf{0.71} \pm .01$	$\mathbf{0.78} \pm .05$	$\mathbf{0.41} \pm .03$

To further characterize out-of-distribution performance, in Figure 3.3 we show how each method's performance on the NIST-20 test split decays as test examples decrease in similarity to the method's respective training set. Specifically, we compute, for each structure in the test set, the maximum Tanimoto similarity between its radius-2 Morgan fingerprint and that of any training structure. While all methods suffer out-of-distribution, GRAFF-MS maintains its edge over the other methods at all but the very highest levels of dissimilarity from NIST-20, where it approaches CFM-ID. CFM-ID's more gradual decay compared to the deep learning methods likely reflects the strong inductive bias of bond-breaking.

3.7.3 Representing peaks as subformulas scales better with molecular weight than substructures

Figure 3.4 shows how our approach to modelling high resolution spectra scales better with input size than bond-breaking. CFM-ID, which is written in optimized C++ code, takes on average 4.9 seconds per structure in the NIST-20 test split, and scales quadratically ($R^2 = 0.78$) with input size. (We believe this is because larger molecules in NIST-20 tend to be approximately path graphs – e.g. long hydrocarbon chains – with only quadratically many connected subgraphs.) In comparison, running our research implementation of GRAFF-MS on the CPU takes 1.3 core-seconds per spectrum, and scales approximately linearly ($R^2 = 0.65$). This pays off at larger molecular weight: for molecules > 500Da, our model is $16\times$ faster on average. Realistically, large-scale library prediction will use the GPU: on

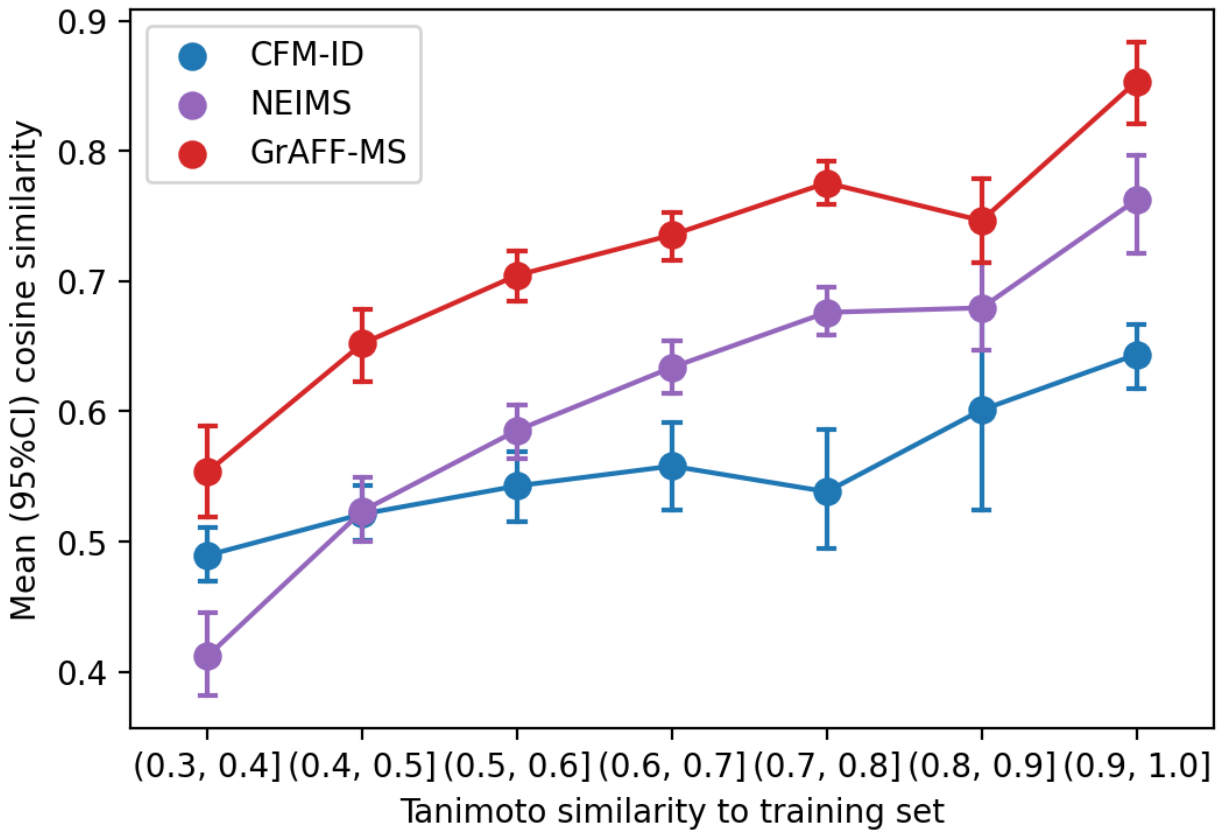

Figure 3.3: Relationship between predictive accuracy on NIST-20 test structures and structural similarity to the training set. Note that CFM-ID uses a different training set (METLIN).

a single GPU with batch size 512, predicting all of the NIST-20 test spectra averages to 2.8ms per spectrum (mostly spent in CPU-bound preprocessing).

3.7.4 GrAFF-MS yields more accurate and faster structure retrieval against a large spectral library

In Table 3.2 we report recall at $k = \{1, 5, 10\}$ of experimental spectra from the NIST-20 test split on our spectral library search task. We see the superior performance of our approach in the spectrum prediction task extends to better downstream retrieval accuracy, in terms of both correct 2D structures and correct chemical formulas. Here the importance of speed at inference time becomes apparent: while both deep learning methods took only a few minutes on a single GPU, CFM-ID required more than 5 node-days of compute time on 8 96-core nodes of an HPC cluster to generate this library.

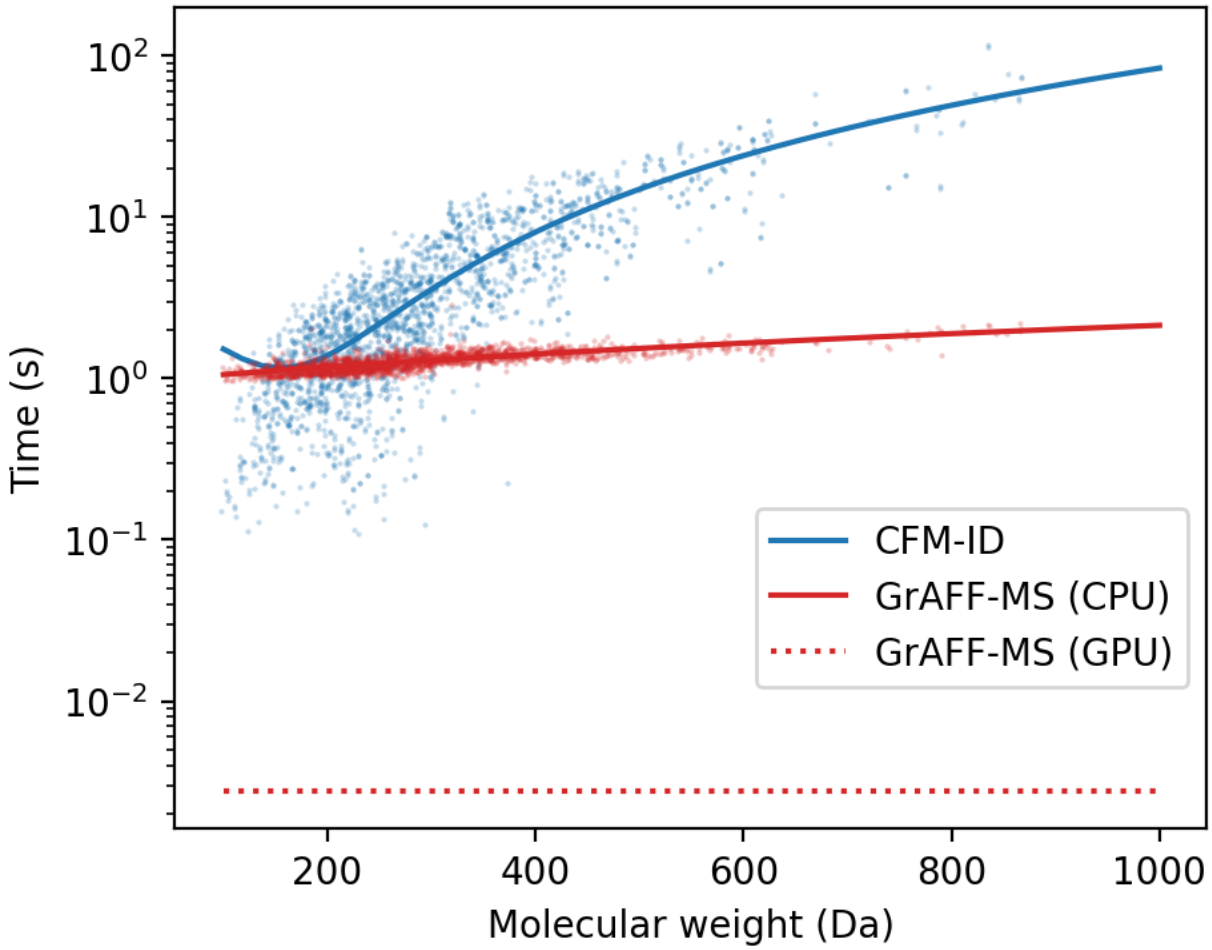

Figure 3.4: Empirical time complexity on NIST-20 structures with respect to molecular weight. Each dot is a structure. Solid lines are quadratic (blue) and linear (red) fits; dotted line indicates an average over all spectra computed using shuffled minibatches.

3.7.5 GrAFF-MS distinguishes very similar compounds and makes human-like mistakes

In Figure 3.5 we show some particularly challenging examples of mass spectra. The top and middle panels show two structurally similar compounds, differing only by the order of one carbon-carbon bond. Our approach correctly predicts distinct spectra for each ($C_{S\hat{S}} = 0.90$, top; $C_{S\hat{S}} = 0.97$, middle). The third molecule is an example where we fail to predict a realistic spectrum ($C_{S\hat{S}} = 0.04$), but in a manner in which a human expert would also fail. This molecule is a member of the *phthalate* class, which chemists recognize by a characteristic dominant peak at 149Da [44]. Our model predicts this same peak, correctly recognizing a phthalate. But in this case that peak is unusually minor – potentially reflecting a long-range dependency in the graph that our approach failed to capture.

Table 3.2: Recall-at-k of NIST-20 test spectra against synthetic libraries predicted from NIST-20 and ChEMBL structures. We report retrieval accuracy of both the correct 2D structure and of any structure with the correct chemical formula, as well as time taken to generate the 1.2 million spectra in the library.

	CFM-ID	NEIMS	GRAFF-MS
STRUCTURE, k=1	0.28±.01	0.27±.01	**0.37**±.01
5	0.56±.02	0.53±.02	**0.67**±.01
10	0.66±.01	0.61±.01	**0.75**±.01
FORMULA, k=1	0.34±.02	0.43±.01	**0.52**±.02
5	0.64±.02	0.65±.01	**0.76**±.01
10	0.74±.01	0.73±.01	**0.83**±.01
INFERENCE TIME	126H7M	**7M32S**	18M52S

3.8 Discussion

In this work, we develop GRAFF-MS, a graph neural network for predicting high resolution mass spectra of small molecules. Unlike previous approaches that force a tradeoff between m/z resolution and a tractable learning problem, GRAFF-MS is both computationally efficient and capable of modelling the high resolution m/z information essential to modern mass spectrometry. This is made possible by our discovery that mass spectra of small molecules can be closely approximated as distributions over a fixed vocabulary of chemical formulas, highlighting the value that domain-aware modelling can add to molecular machine learning. Particularly surprising was that we outperform CFM-ID, which trades model expressivity for an even stronger scientific prior that we expected would contribute to better generalization. However, this prior incurs a heavy cost in time complexity, making it impractical to train CFM-ID on hundreds of thousands of spectra as we did.

While a fixed vocabulary of fragments yields an architecture that is simple to train and fast at inference time, it is possible that this can at times sacrifice flexibility: we may fail to capture fragments of intermediate size that can arise from complex small molecules such as natural products. We believe dynamic generation of this vocabulary, and the related problem of learning generalizable formula representations, to be promising avenues for future work.

Overall we anticipate GRAFF-MS will both accelerate scientific discovery and demonstrate mass spectrometry as a compelling domain for further machine learning research.

Software and Data

We provide code, data, and trained models at https://github.com/murphy17/graff-ms. The NIST-20 license agreement prohibits including spectra from it; we therefore provide instructions on how to obtain it.

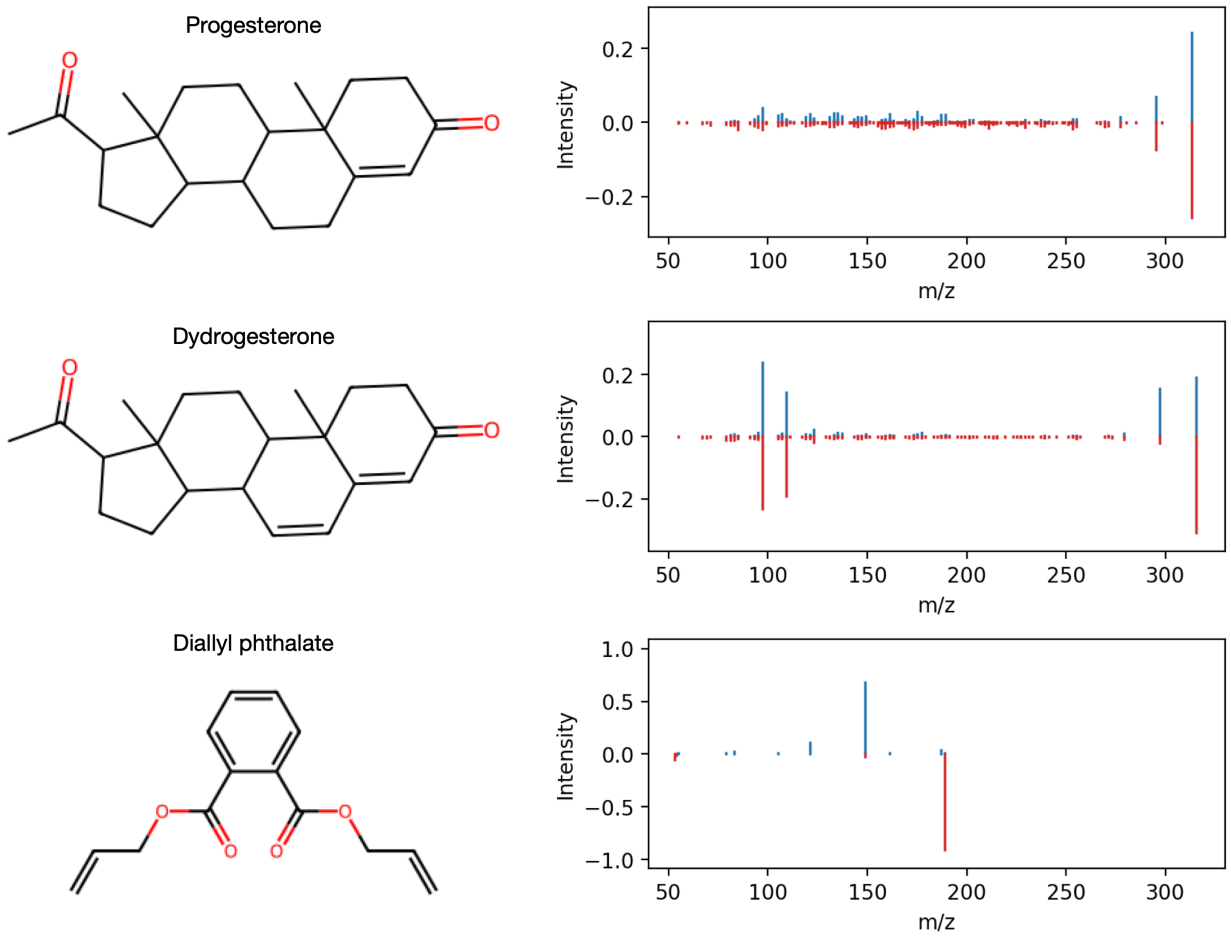

Figure 3.5: Three compounds from CASMI-16, with spectra predicted by our model (blue) against negated ground-truth (red). Oxygens are shaded red by convention.

Acknowledgements

The authors thank the reviewers for their constructive suggestions, and members of the Jegelka and Fraenkel labs, Gennady Voronov, Sam Goldman, and Connor Coley for helpful conversations. The authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing HPC resources that have contributed to the research results reported in this paper. M.M. thanks Enveda Biosciences, the Natural Sciences and Engineering Research Council of Canada, and the Chan-Zuckerberg Initiative for financial support.

3.9 Appendix

3.9.1 Fixed vocabulary selection

Algorithm 2 describes our procedure for selecting the product ions $\hat{\mathcal{P}}$ and neutral losses $\hat{\mathcal{L}}$. We use the shorthand $\mathcal{F}_i^n = \mathcal{F}(P_n, m_i, \epsilon)$ to indicate the set of formulas computed by product mass decomposition. When this yields more than one formula annotation for a peak, here we split the peak height uniformly among all annotations.

Algorithm 2 Fixed vocabulary selection

Input: training spectra and precursors $\{(S_n, P_n)\}_{n=1}^N$, vocabulary size K, tolerance ϵ
Output: product vocabulary $\hat{\mathcal{P}}$, loss vocabulary $\hat{\mathcal{L}}$
Initialize hash-tables $\Pi, \Lambda : \Pi(\cdot) = 0, \Lambda(\cdot) = 0$
Initialize sets $\hat{\mathcal{P}} = \emptyset, \hat{\mathcal{L}} = \emptyset$
for $n = 1 \dots N$ **do**
  for $(m_i, y_i) \in S_n$ **do**
    Compute mass decomposition $\mathcal{F}_i^n = \mathcal{F}(P_n, m_i, \epsilon)$
    for $f \in \mathcal{F}_i^n$ **do**
      $l = P_n - f$
      $\Pi(f) \leftarrow \Pi(f) + y_i/|\mathcal{F}_i^n|$
      $\Lambda(l) \leftarrow \Lambda(l) + y_i/|\mathcal{F}_i^n|$
    end for
  end for
end for
Sort Π and Λ in descending value order
while $|\hat{\mathcal{P}}| + |\hat{\mathcal{L}}| \leq K$ **do**
  $f =$ first element of Π
  $l =$ first element of Λ
  if $\Pi(f) > \Lambda(l)$ **then**
    Add f to $\hat{\mathcal{P}}$
    Remove f from Π
  else
    Add l to $\hat{\mathcal{L}}$
    Remove l from Λ
  end if
end while

(In our implementation, we do not actually compute the mass decomposition in the loop: we instead simply read off the annotations provided already by NIST.)

3.9.2 Derivation of peak-marginal cross entropy

We derive our loss function from physical first principles, making a number of minor modelling assumptions:

- The number of precursor ions accumulated in a spectrum is Poisson with rate λ.
- Each individual precursor ion is independently converted into fragment j with probability p_j.
- The instrument resolution parameter ϵ is sufficiently small that separate peaks do not overlap: there exists exactly one peak $i(j)$ for every $j : p_j > 0$ satisfying $|\langle \mu, f_j \rangle - m_i| \leq \epsilon m_i$ (where f_j is the chemical formula of fragment j).

By the splitting property, the number of ions of each fragment are independently Poisson with rate $\lambda_j = \lambda p_j$. By the merging property, the height of peak i is also a Poisson r.v. K_i with rate $\lambda_i = \sum_{j \in \mathcal{J}_i} \lambda_j$, where $\mathcal{J}_i$ denotes the set of fragments whose theoretical masses fall within the measurement error ϵm_i of peak i. The log-likelihood of peak height is (taking equality up to constants C w.r.t. p_j):

$$\begin{aligned}
&\log P(K_i = k_i) && (3.22)\\
&= k_i \log \lambda_i - \lambda_i - \log k_i! && (3.23)\\
&= k_i \log \left(\sum_{j \in \mathcal{J}_i} \lambda_j \right) - \left(\sum_{j \in \mathcal{J}_i} \lambda_j \right) + C && (3.24)\\
&= k_i \log \left(\sum_{j \in \mathcal{J}_i} \lambda p_j \right) - \left(\sum_{j \in \mathcal{J}_i} \lambda p_j \right) && (3.25)\\
&= k_i \log \lambda + k_i \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) - \lambda \sum_{j \in \mathcal{J}_i} p_j && (3.26)\\
&= C + k_i \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) - \lambda \sum_{j \in \mathcal{J}_i} p_j && (3.27)
\end{aligned}$$

where (3.24) uses merging, and (3.25) uses splitting. Because each product ion is assigned to exactly one peak (no overlap), the peak heights $\{K_i : i \in S\}$ are independent. Let the

total number of accumulated ions $K = \sum_{i \in S} k_i$ in spectrum S. Defining $y_i = k_i / K$:

$$\log P(\{K_i = k_i : i \in S\}) \tag{3.28}$$

$$= \sum_{i \in S} \log P(K_i = k_i) \tag{3.29}$$

$$= \sum_{i \in S} \left(k_i \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) - \lambda \sum_{j \in \mathcal{J}_i} p_j \right) \tag{3.30}$$

$$= \sum_{i \in S} (K y_i) \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) - \lambda \sum_{i \in S} \sum_{j \in \mathcal{J}_i} p_j \tag{3.31}$$

$$= K \sum_{i \in S} y_i \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) - \lambda \cdot 1 \tag{3.32}$$

$$= C \sum_{i \in S} y_i \log \left(\sum_{j \in \mathcal{J}_i} p_j \right) + C' \tag{3.33}$$

where (3.32) again uses our assumption that every fragment is assigned to exactly one peak. Dropping the constants and negating the final term yields the peak-marginal cross-entropy loss for a single spectrum.

3.9.3 Model hyperparameters

We use a vocabulary of $K = 10{,}000$ formulas. We train an $L = 6$-layer encoder and $L' = 2$-layer decoder with $d_{enc} = 512$ and $d_{dec} = 1024$, resulting in 24.1 million trainable parameters. We use the $d_{eig} = 8$ lowest-frequency eigenvalues, truncating or padding with zeros. Dropout is applied at rate 0.1. We use a batch size of 512 and the Adam optimizer [45] with learning rate 5×10^{-4} and weight decay 10^{-5}. We train for 100 epochs and use the model from the epoch with the lowest validation loss. All models are trained using PyTorch Lightning with automatic mixed precision on 2 Tesla V100 GPUs.

3.9.4 Mass spectral covariates

Table 3.3: Mass spectral covariates used in our model.

Feature	Range	Comment
Collision energy	$[0, 200]$	Thermo Scientific PSB104, "Normalized Collision Energy Technology"
Precursor type	$[M+H]^+$, $[M-H]^-$	Includes ionization mode & adduct composition
Instrument model	Orbitrap Fusion Lumos, Thermo Finnigan Elite Orbitrap, Thermo Finnigan Velos Orbitrap	Different limits of detection
Has isotopic peaks	False, True	Proxy for width setting of precursor mass filter

References

[1] Katja Dettmer, Pavel A. Aronov, and Bruce D. Hammock. "Mass spectrometry-based metabolomics". In: *Mass Spectrometry Reviews* 26.1 (Aug. 2006), pp. 51–78. DOI: 10.1002/mas.20108. URL: https://doi.org/10.1002/mas.20108.

[2] Atanas G Atanasov et al. "Natural products in drug discovery: advances and opportunities". In: *Nature reviews Drug discovery* 20.3 (2021), pp. 200–216.

[3] Ethan D. Evans et al. "Predicting human health from biofluid-based metabolomics using machine learning". In: *Scientific Reports* 10.1 (Oct. 2020). DOI: 10.1038/s41598-020-74823-1. URL: https://doi.org/10.1038/s41598-020-74823-1.

[4] Hilary M. Brown et al. "The current role of mass spectrometry in forensics and future prospects". In: *Analytical Methods* 12.32 (2020), pp. 3974–3997. DOI: 10.1039/d0ay01113d. URL: https://doi.org/10.1039/d0ay01113d.

[5] F. Hernandez et al. "Current use of high-resolution mass spectrometry in the environmental sciences". In: *Analytical and Bioanalytical Chemistry* 403.5 (Feb. 2012), pp. 1251–1264. DOI: 10.1007/s00216-012-5844-7. URL: https://doi.org/10.1007/s00216-012-5844-7.

[6] Ricardo R da Silva, Pieter C Dorrestein, and Robert A Quinn. "Illuminating the dark matter in metabolomics". In: *Proceedings of the National Academy of Sciences* 112.41 (2015), pp. 12549–12550.

[7] Oliver Fiehn. *Critical Assessment of Small Molecule Identification 2022*. http://www.casmi-contest.org/2022/index.shtml/. [Online; accessed 12-December-2022]. 2022.

[8] Stephen Stein. "Mass Spectral Reference Libraries: An Ever-Expanding Resource for Chemical Identification". In: *Analytical Chemistry* 84.17 (July 2012), pp. 7274–7282. DOI: 10.1021/ac301205z. URL: https://doi.org/10.1021/ac301205z.

[9] Christoph A Krettler and Gerhard G Thallinger. "A map of mass spectrometry-based in silico fragmentation prediction and compound identification in metabolomics". In: *Briefings in Bioinformatics* 22.6 (Mar. 2021). DOI: 10.1093/bib/bbab073. URL: https://doi.org/10.1093/bib/bbab073.

[10] Jennifer N. Wei et al. "Rapid Prediction of Electron-Ionization Mass Spectrometry Using Neural Networks". In: *ACS Central Science* 5.4 (Mar. 2019), pp. 700–708. DOI: 10.1021/acscentsci.9b00085. URL: https://doi.org/10.1021/acscentsci.9b00085.

[11] Hao Zhu, Liping Liu, and Soha Hassoun. *Using Graph Neural Networks for Mass Spectrometry Prediction*. 2020. DOI: 10.48550/ARXIV.2010.04661. URL: https://arxiv.org/abs/2010.04661.

[12] Adamo Young, Bo Wang, and Hannes Rost. *MassFormer: Tandem Mass Spectrum Prediction with Graph Transformers*. 2021. DOI: 10.48550/ARXIV.2111.04824. URL: https://arxiv.org/abs/2111.04824.

[13] Fei Wang et al. "CFM-ID 4.0: More Accurate ESI-MS/MS Spectral Prediction and Compound Identification". In: *Analytical Chemistry* 93.34 (Aug. 2021), pp. 11692–11700. DOI: 10.1021/acs.analchem.1c01465. URL: https://doi.org/10.1021/acs.analchem.1c01465.

[14] Christoph Ruttkies, Steffen Neumann, and Stefan Posch. "Improving MetFrag with statistical learning of fragment annotations". In: *BMC Bioinformatics* 20.1 (July 2019). DOI: 10.1186/s12859-019-2954-7. URL: https://doi.org/10.1186/s12859-019-2954-7.

[15] Kai Duhrkop et al. "Faster Mass Decomposition". In: *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 45–58. DOI: 10.1007/978-3-642-40453-5_5. URL: https://doi.org/10.1007/978-3-642-40453-5_5.

[16] Tobias Kind and Oliver Fiehn. "Seven Golden Rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry". In: *BMC Bioinformatics* 8.1 (Mar. 2007). DOI: 10.1186/1471-2105-8-105. URL: https://doi.org/10.1186/1471-2105-8-105.

[17] Felicity Allen, Russ Greiner, and David Wishart. "Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification". In: *Metabolomics* 11.1 (June 2014), pp. 98–110. DOI: 10.1007/s11306-014-0676-4. URL: https://doi.org/10.1007/s11306-014-0676-4.

[18] Liu Cao et al. "MolDiscovery: learning mass spectrometry fragmentation of small molecules". In: *Nature Communications* 12.1 (2021), pp. 1–13.

[19] Anna Gaulton et al. "The ChEMBL database in 2017". In: *Nucleic Acids Research* 45.D1 (Nov. 2016), pp. D945–D954. DOI: 10.1093/nar/gkw1074. URL: https://doi.org/10.1093/nar/gkw1074.

[20] F. W. McLafferty. "Mass Spectrometric Analysis. Molecular Rearrangements". In: *Analytical Chemistry* 31.1 (Jan. 1959), pp. 82–87. DOI: 10.1021/ac60145a015. URL: https://doi.org/10.1021/ac60145a015.

[21] Jeroen Koopman and Stefan Grimme. "From QCEIMS to QCxMS: A Tool to Routinely Calculate CID Mass Spectra Using Molecular Dynamics". In: *Journal of the American Society for Mass Spectrometry* 32.7 (June 2021), pp. 1735–1751. DOI: 10.1021/jasms.1c00098. URL: https://doi.org/10.1021/jasms.1c00098.

[22] Xie-Xuan Zhou et al. "pDeep: Predicting MS/MS Spectra of Peptides with Deep Learning". In: *Analytical Chemistry* 89.23 (Nov. 2017), pp. 12690–12697. DOI: 10.1021/acs.analchem.7b02566. URL: https://doi.org/10.1021/acs.analchem.7b02566.

[23] Siegfried Gessulat et al. "Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning". In: *Nature Methods* 16.6 (May 2019), pp. 509–518. DOI: 10.1038/s41592-019-0426-7. URL: https://doi.org/10.1038/s41592-019-0426-7.

[24] Richard Licheng Zhu and Eric Jonas. "Rapid Approximate Subset-Based Spectra Prediction for Electron Ionization–Mass Spectrometry". In: *Analytical Chemistry* 95.5 (Jan. 2023), pp. 2653–2663. DOI: 10.1021/acs.analchem.2c02093. URL: https://doi.org/10.1021/acs.analchem.2c02093.

[25] Samuel Goldman, Janet Li, and Connor W. Coley. *Generating Molecular Fragmentation Graphs with Autoregressive Neural Networks*. 2023. DOI: 10.48550/ARXIV.2304.13136. URL: https://arxiv.org/abs/2304.13136.

[26] Keyulu Xu et al. "How Powerful are Graph Neural Networks?" In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL: https://openreview.net/forum?id=ryGs6iA5Km.

[27] Weihua Hu et al. "Strategies for Pre-training Graph Neural Networks". In: *International Conference on Learning Representations*. 2020. URL: https://openreview.net/forum?id=HJlWWJSFDH.

[28] Derek Lim et al. *Sign and Basis Invariant Networks for Spectral Graph Representation Learning*. 2022. DOI: 10.48550/ARXIV.2202.13013. URL: https://arxiv.org/abs/2202.13013.

[29] Justin Gilmer et al. “Neural Message Passing for Quantum Chemistry”. In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1263–1272. URL: http://proceedings.mlr.press/v70/gilmer17a.html.

[30] Mufei Li et al. “DGL-LifeSci: An Open-Source Toolkit for Deep Learning on Graphs in Life Science”. In: *ACS Omega* (2021).

[31] Matthias Fey and Jan E. Lenssen. “Fast Graph Representation Learning with PyTorch Geometric”. In: *ICLR Workshop on Representation Learning on Graphs and Manifolds*. 2019.

[32] Tianle Cai et al. “GraphNorm: A Principled Approach to Accelerating Graph Neural Network Training”. In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 1204–1215. URL: http://proceedings.mlr.press/v139/cai21e.html.

[33] Meng Joo Er et al. “Attention pooling-based convolutional neural network for sentence modelling”. In: *Information Sciences* 373 (Dec. 2016), pp. 388–403. DOI: 10.1016/j.ins.2016.08.084. URL: https://doi.org/10.1016/j.ins.2016.08.084.

[34] National Institute of Standards and Technology. *NIST-20*. 2020. URL: https://www.nist.gov/programs-projects/nist20-updates-nist-tandem-and-electron-ionization-spectral-libraries.

[35] Kai Duhrkop. “Deep kernel learning improves molecular fingerprint prediction from tandem mass spectra”. In: *Bioinformatics* 38.Supplement 1 (June 2022), pp. i342–i349. DOI: 10.1093/bioinformatics/btac260. URL: https://doi.org/10.1093/bioinformatics/btac260.

[36] Stephen R Heller et al. “InChI, the IUPAC International Chemical Identifier”. In: *Journal of Cheminformatics* 7.1 (May 2015). DOI: 10.1186/s13321-015-0068-4. URL: https://doi.org/10.1186/s13321-015-0068-4.

[37] Zhenqin Wu et al. “MoleculeNet: a benchmark for molecular machine learning”. In: *Chemical Science* 9.2 (2018), pp. 513–530. DOI: 10.1039/c7sc02664a. URL: https://doi.org/10.1039/c7sc02664a.

[38] Emma L. Schymanski et al. “Critical Assessment of Small Molecule Identification 2016: automated methods”. In: *Journal of Cheminformatics* 9.1 (Mar. 2017). DOI: 10.1186/s13321-017-0207-1. URL: https://doi.org/10.1186/s13321-017-0207-1.

[39] Mingxun Wang et al. “Sharing and community curation of mass spectrometry data with Global Natural Products Social Molecular Networking”. In: *Nature biotechnology* 34.8 (2016), pp. 828–837.

[40] Carlos Guijas et al. “METLIN: A Technology Platform for Identifying Knowns and Unknowns”. In: *Analytical Chemistry* 90.5 (Jan. 2018), pp. 3156–3164. DOI: 10.1021/acs.analchem.7b04424. URL: https://doi.org/10.1021/acs.analchem.7b04424.

[41] Maria Lorna A. De Leoz et al. "Cross-Ring Fragmentation Patterns in the Tandem Mass Spectra of Underivatized Sialylated Oligosaccharides and Their Special Suitability for Spectrum Library Searching". In: *Journal of the American Society for Mass Spectrometry* 30.3 (Dec. 2018), pp. 426–438. DOI: 10.1007/s13361-018-2106-8. URL: https://doi.org/10.1007/s13361-018-2106-8.

[42] Stephen E. Stein and Donald R. Scott. "Optimization and testing of mass spectral library search algorithms for compound identification". In: *Journal of the American Society for Mass Spectrometry* 5.9 (Sept. 1994), pp. 859–866. DOI: 10.1016/1044-0305(94)87009-8. URL: https://doi.org/10.1016/1044-0305(94)87009-8.

[43] Florian Huber et al. "matchms - processing and similarity evaluation of mass spectrometry data." In: *Journal of Open Source Software* 5.52 (Aug. 2020), p. 2411. DOI: 10.21105/joss.02411. URL: https://doi.org/10.21105/joss.02411.

[44] Yassin A. Jeilani, Beatriz H. Cardelino, and Victor M. Ibeanusi. "Density Functional Theory and Mass Spectrometry of Phthalate Fragmentations Mechanisms: Modeling Hyperconjugated Carbocation and Radical Cation Complexes with Neutral Molecules". In: *Journal of the American Society for Mass Spectrometry* 22.11 (Aug. 2011). DOI: 10.1007/s13361-011-0215-8. URL: https://doi.org/10.1007/s13361-011-0215-8.

[45] Diederik P Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: (2015).

Chapter 4

Comparing single-cell sequencing platforms for gene regulatory network inference

This section of the thesis describes work I carried out as a member of the Standards and Technologies Working Group (STWG) of the Human Cell Atlas. These results will be included in a manuscript under preparation, jointly authored by the STWG. Helpful discussions were had with Ernest Fraenkel and Stefanie Jegelka.

Abstract

Single-cell omics measurements present the possibility of statistically inferring the topology and parameters of gene regulatory networks. However, the quality of such inferences depends heavily upon the biomolecular variables measured and technical biases induced by the particular choice of measurement technology. In this work, we consider three different single-cell sequencing platforms – single-cell RNA sequencing, single-nucleus multiome sequencing, and single-cell CAGE sequencing – measured in a complex tissue (human kidney cortex), and compare them along three different axes: single-cell versus single-nucleus isolation; transcriptomic versus multimodal measurements; and chromatin accessibility versus promoter usage as epigenomic measurements. We adopt various resampling strategies to control for technical variation in cell-type biases and read depth, and use elastic-net linear models as a fast and interpretable approach to inferring regulatory networks that can be compared across different measurement modalities. We employ a number of auxiliary external datasets for hypothesis generation and quantitative ground-truth evaluation. Among our findings, we observe evidence of subcellular mRNA localization influencing statistics of single-cell versus single-nuclear expression; superior predictability of the latter relative to the former; and a general trade-off between biological plausibility and predictive power of inferred regulatory networks, both with and without epigenomic constraints.

4.1 Introduction

Contemporary omics sequencing experiments can measure tens to hundreds of thousands of molecular variables in thousands of single cells. Measurements of this scale raise the possibility of inferring topologies and parameters of biological regulatory networks with greater throughput and less bias than classical low-dimensional biochemical approaches. As such networks are heavily tissue-specific [1], the completion of the Human Cell Atlas (HCA) [2] project – which aims to acquire broad and deep single-cell sequencing data in all major human organs – will be an important step toward this goal.

A major experimental design consideration in construction of this Atlas is the selection of which particular sequencing assay, and combinations thereof, to carry out. Different single-cell assays can measure different classes of molecular variables – e.g. gene expression [3], chromatin accessibility [4], surface protein expression [5] – either separately or jointly [6], and entail different sample preparation protocols that introduce sampling biases and expression artifacts. Toward this end, the Standards and Technologies Working Group (STWG) of the HCA consortium was established to better understand and quantify the technical biases and complementarity of a number of single-cell sequencing assays under consideration for the Atlas. A major aspect of this investigation – with which the author of this thesis was tasked – was to understand how regulatory networks inferred from these assays compare to one another and to biological ground-truth.

This is a challenging task for a number of reasons: firstly, it necessitates establishing a common algorithmic approach and comparison methodology for inferring and evaluating networks derived from measurements spanning heterogeneous measurement modalities. We also need to establish clear hypotheses to investigate, and must take care to separate potential biological effects of interest from already well-studied technical biases. Finally, we must deal with a problem pervasive to biological machine learning: ground-truth gene regulatory networks in complex tissues are simply not known at scale, so we must rely upon domain knowledge and external datasets to devise metrics for evaluation.

4.2 Background

Both the assays directly performed by the Standards and Technologies Working Group, and the external data sources we use to validate them, span a diverse range of modalities and experimental protocols. We briefly explain each of these assays in this section.

4.2.1 Single-cell RNA sequencing

Due to its relative technical ease and commercial maturity, the predominant approach for high-throughput molecular characterization of biological systems is single-cell RNA sequencing [3]. Briefly, this technology begins with physical isolation of single cells or nuclei into microfluidic droplets, and lyses them to extract the RNA content. Individual RNA transcripts are then reverse-transcribed into DNA, which includes barcoding with a unique molecule identifier (UMI) and a cell identifier, and amplified via PCR. An illustration is shown in Figure 4.1.

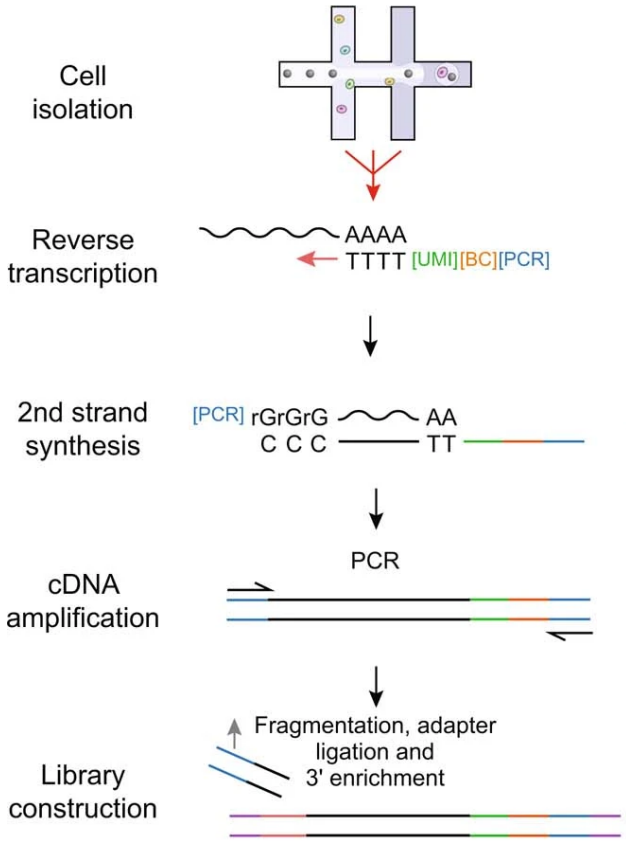

Figure 4.1: Single-cell RNA sequencing. Following tissue dissociation, single cells or nuclei are isolated into microfluidic droplets, and their RNA content is extracted. Individual RNA molecules are hybridized to probe sequences, here complementary to the 3'-end poly-A tail, that uniquely identify the molecule and cell, and reverse-transcribed to DNA. The complementary DNA fragments are amplified by PCR, and then read out by a sequencer. (Figure from [7].)

The full transcript is not generally sequenced in high-throughput single-cell RNA experiments: the easiest approach sequences only the 3' end of transcripts possessing a poly-A tail. 5'-end sequencing [8] is more technically difficult, and relies on capturing the methylated guanine 'cap' at the 5'-end of mRNA transcripts. However, it provides the benefit of directly reading the sequence of the transcription start site. Full-length sequencing [9] permits characterizing alternate promoter usage and splicing isoforms, but achieves much lower cell throughput [7].

Different experimental design choices in the single-cell RNA workflow are known to induce particular technical biases. Single-cell versus single-nuclear sequencing entail different sample preparation protocols: enzymatic dissociation of a tissue into single cells induces global stress response [7], and tends to damage certain cell subpopulations more than others [10]. In comparison, single-nucleus isolation is easier technically and does not induce this stress response: however, it excludes cytoplasmic transcripts, which constitute the majority, and depletes specific cell types as well [10]. As polyadenylation rates vary across genes, the choice

of 3'- versus 5'-end sequencing can also induce sampling bias [11].

It is likely that technical differences in cell type recovery and induced stress between single-cell and single-nuclear sequencing will lead to differences in inferred gene regulatory networks. However, biological factors could also be involved: cytoplasmic versus nuclear localization of mRNAs can vary greatly across genes [12, 13, 14] and enriches for various regulatory processes [15, 16]. In the nucleus, the rate at which transcripts are exported varies widely across different genes [17, 18], as selective export of already-transcribed mRNAs permits faster translational response to stimuli than needing to wait for transcription first [12]. In the cytoplasm, localization of mRNA transcripts physically near to the regions of the cell where their protein products are needed permits fast translational response by free-floating ribosomes [19]. Additionally, RNA instability is known [20] to enrich for certain biological processes – primarily global regulatory processes, such as RNA splicing, transcriptional regulation, and ribosome biogenesis – for which fast response is evolutionarily favorable, and differential half-lives in the nucleus versus the cytosol will affect the steady-state concentrations of mRNA in each [16]. Depending on depth of sequencing, sampling primarily cytoplasmic versus nuclear mRNA pools could therefore risk undersampling genes subject to these forms of regulation.

Furthermore, the properties of expression noise in one fraction versus the other also differ. Battich et al [21] indicate nuclear transcript counts are overdispersed relative to cytoplasmic counts of the same genes: after controlling for global cell state variables with slower timescales than transcriptional bursting, they find the variance of cytoplasmic transcript abundance approaches the theoretical limit under a Poisson process. They attributed this variance reduction to smoothing brought about by nuclear buffering of transcriptional bursts. As the experimental design of [21] only measured a single gene at a time within any given cell, they only consider the effect of nuclear buffering on expression variance. However, it is conceivable that the smoothing brought about by nuclear buffering could attenuate not just independent noise, but also correlated fluctuations between different genes. As we rely on such correlations to detect statistical dependence, it may be the case that RNA measurements derived primarily from the cytoplasm (i.e. single cell RNA sequencing) have less power per read to detect gene co-regulation than those selected from solely the nucleus, which are spatially and temporally closer to the events of transcription.

4.2.2 Single-cell multiome sequencing

Single-cell RNA sequencing is a powerful approach for identifying cell types and states, and their characteristic gene expression patterns. However, when used as a technology for inference of transcriptional regulatory networks, it suffers fundamental limitations. Ideally, each directed edge in a gene regulatory network would reflect the process illustrated in Figure 4.2: the binding of a protein to a region of DNA that proximally causes (e.g. by directly participating in a transcription preinitiation complex, or physically blocking one from forming) an increase or decrease in the rate of transcription of the target. This is valuable both for scientific interpretability, and because it facilitates the confirmation or falsification of the existence of inferred edges via experimental intervention.

However, single-cell RNA sequencing only directly measures *one* of the molecular variables in this process – the abundance of the target transcript – and moreover only as a

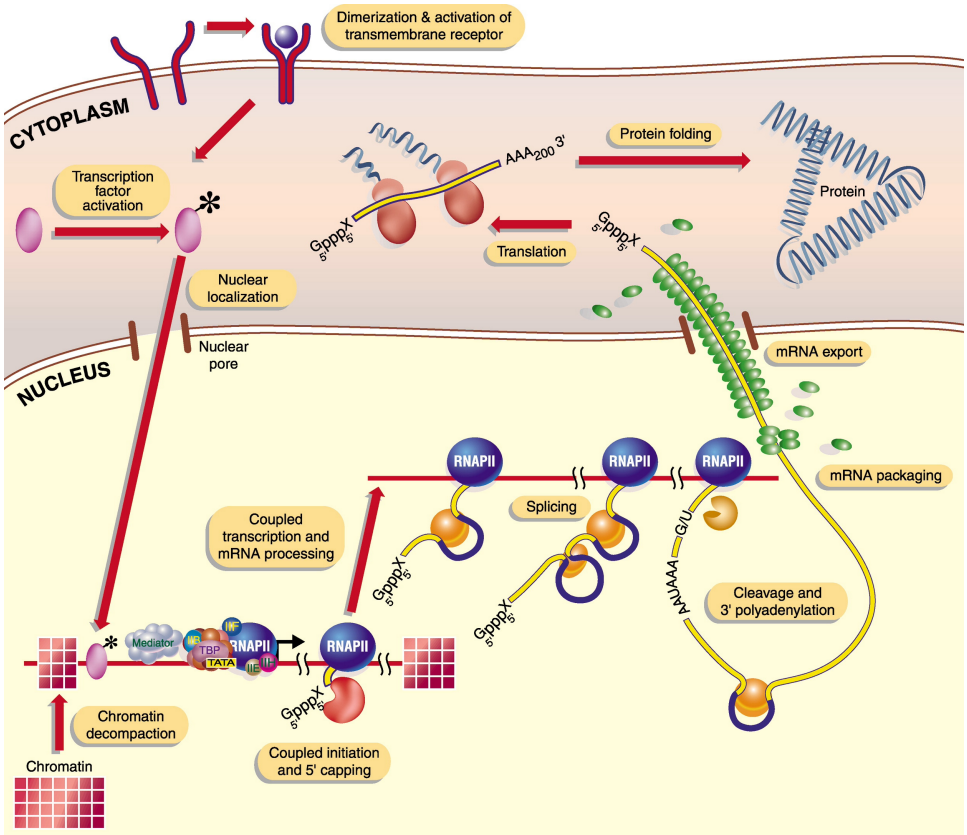

Figure 4.2: The ultimate cause of a transcriptional event is the binding of a ligand to a surface or intracellular receptor, which initiates a signalling cascade ending in the post-translation modification of a transcription factor. This activated TF translocates to the nucleus (if not already localized there), and binds to an accessible region of DNA to which its pocket is complementary. If this site is physically close to a gene it has evolved to regulate, this binding will cause a transcription preinitiation complex to form at the promoter region of the gene. This complex includes RNA polymerase II, which transcribes the RNA molecule. Following initiation of the transcriptional event, numerous other processes influence the lifespan of the nascent RNA transcript, including 5'-end capping, 3'-end polyadenylation, nuclear export, ribonucleoprotein complex formation, and degradation in the cytoplasm. (Figure from [22].)

population, with different transcripts resulting from potentially many different transcriptional events, which are also subject to post-transcriptional regulation via mRNA splicing, export, and degradation. Unfortunately, an assay that can directly observe these microscopic events at high throughput remains out of reach. While regulatory networks can nonetheless be inferred from single-cell RNA sequencing data [23, 24], the fact that important molecular variables remain latent risks detecting spurious or indirect correlations in addition to direct physical mechanisms.

Multimodal single-cell measurements, which observe additional molecular variables involved in gene expression, can in principle aid in ruling out spurious connections between transcription factors and their putative targets. A commercially mature example of such an assay is single-nucleus multiome sequencing [25]: in addition to quantifying nuclear tran-

scripts, multiome also uses 'assay for transposase-accessible chromatin' (ATAC) sequencing to detect regions of DNA that are 'open' – i.e. not wound around histone proteins – and hence physically available for transcription factors to bind to. The principle of this assay is depicted in Figure 4.3; briefly, each cell's DNA is isolated, and exposed to an enzyme that transcribes complementary DNA from regions of open chromatin. These short DNA sequences are aligned to the genome, and aggregated across cells into longer 'peaks': the final measurement is a count-valued matrix of cells by peaks.

As there are only two copies of each chromosome in most eukaryotic cells, single-nucleus ATAC data tends to be much sparser than RNA sequencing. It moreover provides no direct information about whether a TF was bound to a particular genomic locus, or what that TF might have been: for this one must rely on indirect proxies, e.g. enrichment of TF binding motifs in accessible peaks [27] or correlation between peak accessibility and gene expression of TFs [28]. Once connections between TFs and peaks have been established, the set of permitted links in a gene regulatory network can then be constrained, e.g. to only those TFs with evidence of binding to peaks near in the genome to a particular gene.

4.2.3 Single-cell CAGE sequencing

Single-cell cap analysis gene expression (CAGE) sequencing is an alternative implementation of single-cell RNA sequencing, which captures and sequences the 5'-end of transcripts, via capture of the 5'-cap, as opposed to the poly-A tail at the 3'-end. While more technically involved to carry out than 3'-end sequencing, it provides the added benefit of reading the sequence of the transcription start site (TSS) that gave rise to each transcript. This importantly permits detection of alternate promoter usage, which is a widespread mechanism of transcriptional control: more than half of human genes undergo regulation by switching of promoters [29].

As each detected transcript is now associated with a promoter sequence, we can consider CAGE-seq as a multimodal assay in its own right, to which we can apply the same tools – e.g. TF motif enrichment – that we use for ATAC-seq analysis, with the added benefit of much higher read counts. It moreover is a more 'causal' measurement than single-nucleus ATAC sequencing: we know that a particular promoter region caused the expression of a detected transcript when we directly measure that region as part of the transcript, rather having to rely on 'guilt by association' of an ATAC peak merely being physically near to the gene or correlated with its expression. It is on the other hand more limited than ATAC-seq, in that it only samples the subset of the accessible genome corresponding to promoter regions, whereas ATAC-seq is in principle unbiased and can capture regulatory binding events far from the transcription start site.

4.2.4 Immunohistochemistry

A major difficulty in biological machine learning is the rarity of ground truth data against which omics measurement technologies and algorithms can be evaluated. Moreover, when such data does exist, it is often acquired through low-throughput molecular biological experiments, which suffer from selection biases relative to high-throughput omics data and typically employ very different measurement technologies, often with their own sources of

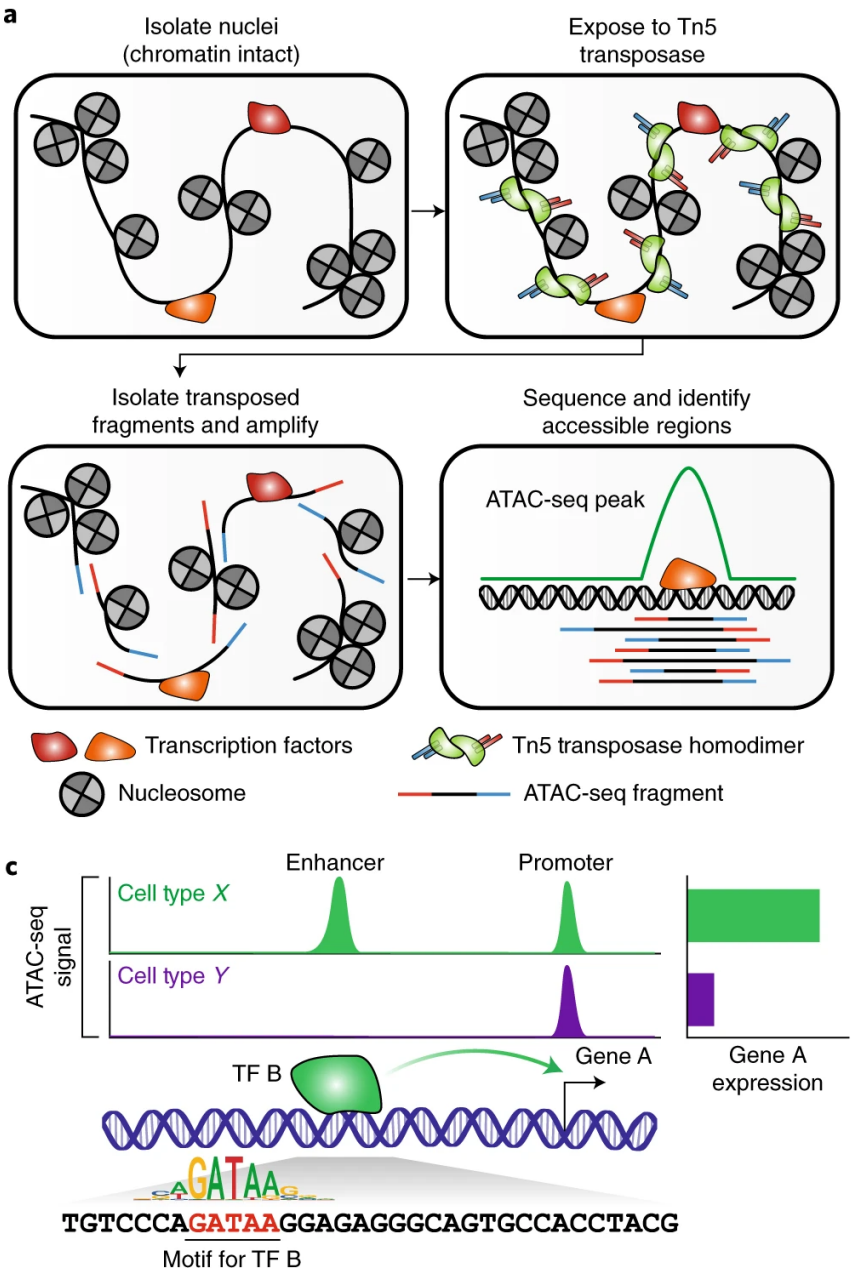

Figure 4.3: Single-nucleus ATAC sequencing isolates the nucleus of each cell, and exposes its DNA content to an enzyme that replicates regions of DNA not spooled around histone proteins. The resulting DNA fragments are amplified, sequenced, and aligned to the genome, and reads are pooled across cells to identify peaks representing accessible loci. Motif enrichment is used to associate each peak with candidate TF binders; if such peaks are located proximally to the transcription start site of a particular gene, it suggests the motif-enriched TF is involved in regulation of that gene. (Figure from [26].)

technical error. In such situations, we must be careful about what we actually define to be 'ground truth' and (in a classification problem – e.g. gene regulatory network inference) what counts as a false positive or a false negative.

This issue notwithstanding, there are basic biophysical principles that – irrespective of measurement technology – we can safely assume correct predictions to satisfy, and which we can make use of in quantitative evaluations. For example, in order for a gene to act as a transcription factor in a particular tissue, we expect that it should not only be transcribed in cells of that tissue – i.e. detected via RNA sequencing – but also translated into a protein and transported back into the nucleus. We might therefore evaluate a particular approach to regulatory network inference in terms of how much 'influence' is assigned to transcription factors with or without evidence of nuclear protein expression in the tissue in question.

A large-scale experimental repository of such evidence can be found in the Human Protein Atlas (HPA) [30]. Among other data types, the HPA includes replicated images of tens of thousands of antibodies, covering much of the measurable proteome, in several major human tissues, imaged via immunohistochemistry (IHC). Immunohistochemistry provides spatially-resolved measurements of protein expression by staining a tissue sample with an antibody complementary to an epitope on a target protein of interest, which is then imaged via optical microscope at sufficient magnification to distinguish subcellular localization patterns. Classically, IHC measures a single protein at a time, in combination with a nucleic acid counterstain: however, multiplexed implementations of the technology have been developed more recently [31]. Examples of immunohistochemistry images are shown in Figure 4.4.

Immunohistochemistry is favored for its relative experimental simplicity; however it is known to suffer from substantial technical variability and artifacts arising from off-target binding of the antibody [32, 33]. To mitigate this issue, the images in the Human Protein Atlas undergo extensive manual curation by expert pathologists, and staining patterns are assess for reproducibility within and between antibodies as well as with respect to bulk RNA sequencing [34]. Importantly, this manual curation process also involves annotation of each antibody for cell-type specificity and subcellular localization (albeit, as discussed elsewhere in this thesis, at relatively coarse scale). As the annotated proteins cover a substantial fraction of the known TF-ome, we can use these data to quantitatively evaluate our gene regulatory inferences.

4.2.5 Chromatin immunoprecipitation sequencing

The nuclear protein localization data found in the Human Protein Atlas permits evaluation of gene regulatory inferences at the level of transcription factors. However, it ignores another condition that our mechanistic picture of transcriptional regulation requires TFs to satisfy: once within the nucleus, they must bind to accessible chromatin in order to modulate the transcription of their targets. An assay that provides this binding information is chromatin immunoprecipitation (ChIP) sequencing [35]. This technology uses an antibody complementary to a transcription factor of interest to specifically precipitate bound DNA-protein complexes containing that TF from a cell lysate. The precipitate is then treated to denature the TF, and the DNA content is extracted, amplified, sequenced, and aligned to the genome. An illustration is provided in Figure 4.5.

ChIP-seq provides valuable direct experimental measurement of links between transcription factors and accessible genomic loci that we can exploit when evaluating inferred regulatory networks. However, the technology is limited to transcription factors for which reliable antibodies compatible with the protocol and system of interest are available. It also indicates

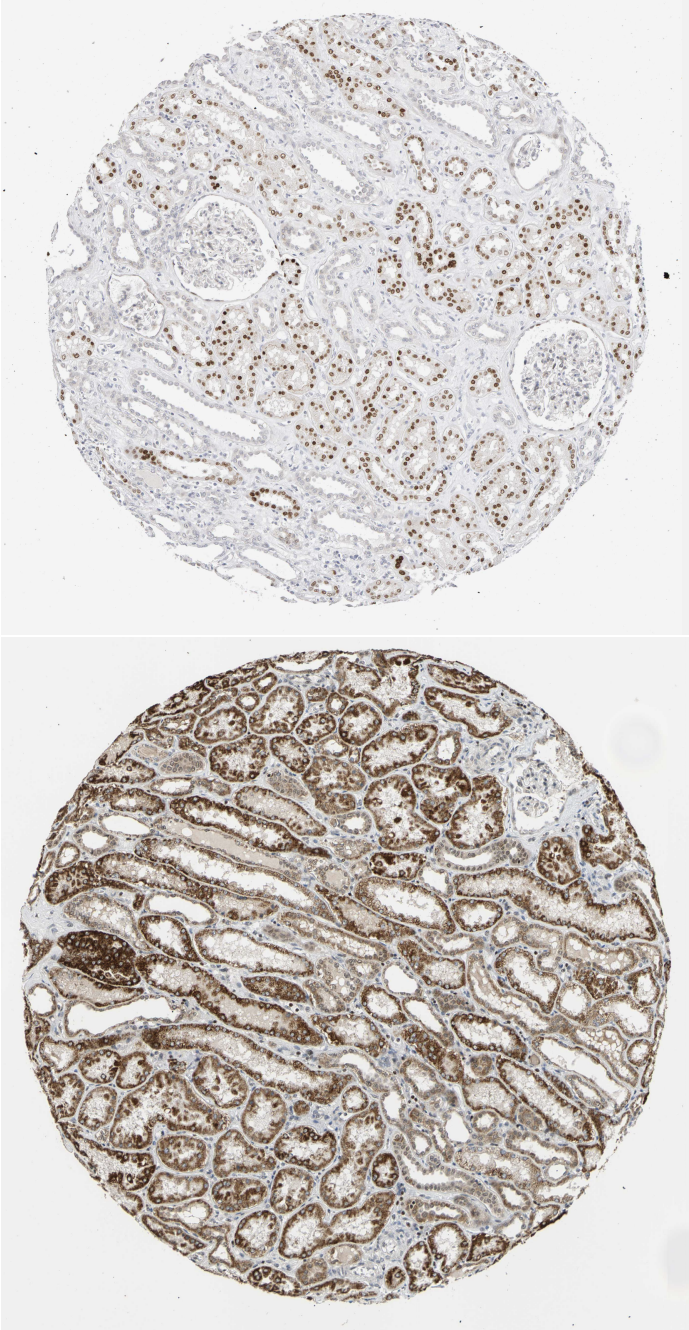

Figure 4.4: Example IHC images from the HPA of two transcription factor proteins in kidney. The antibody stain is brown, and the nucleic acid counterstain is blue. Top, HNF4A displays nuclear localization in cells of the proximal tubule, suggesting it is active; bottom, SP4 is highly expressed but localizes only to the cytoplasm – indicating the protein (at least in the conformation targeted by the selected antibody) is not physically participating in transcription.

nothing about which regulatory targets are being modulated by those bound TFs.

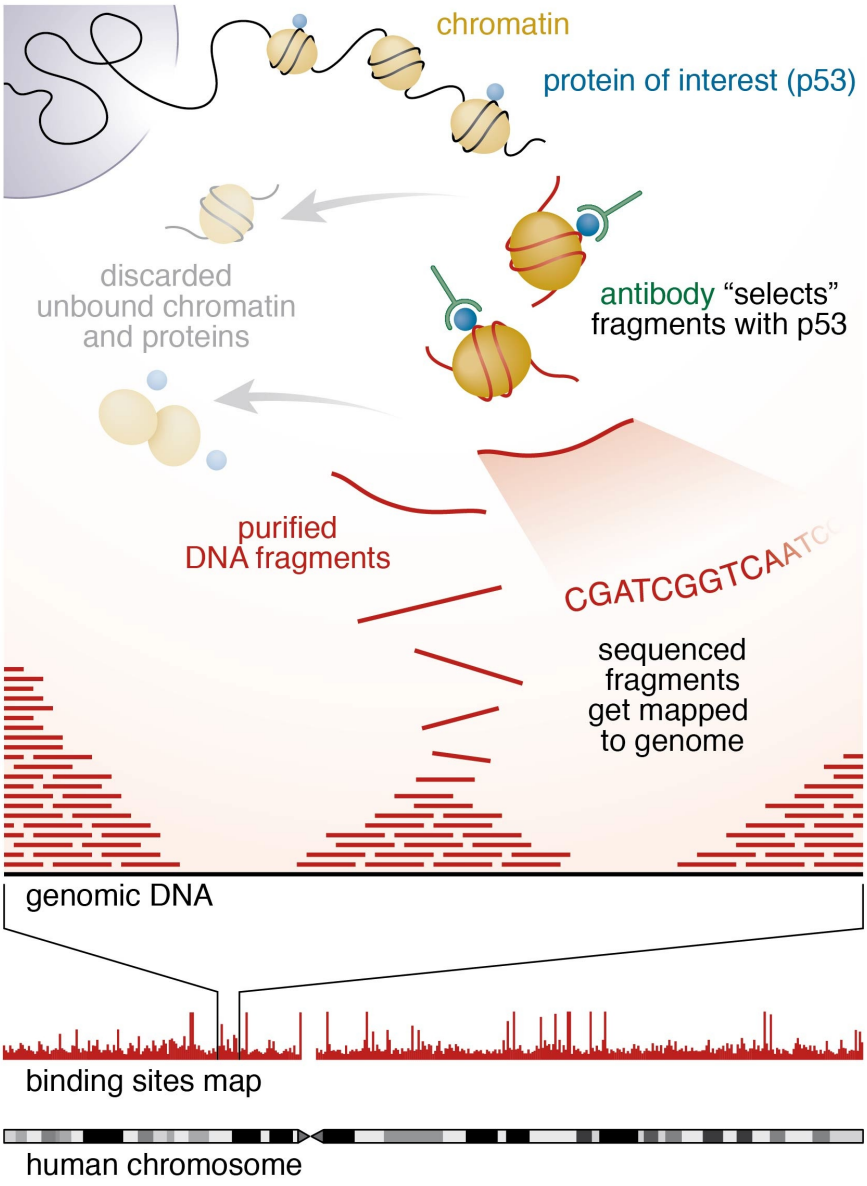

Figure 4.5: Chromatin immunoprecipitation (ChIP) sequencing uses an antibody complementary to a protein of interest to extract DNA bound to that protein. The DNA is purified, amplified, and sequenced. The resulting fragments, when mapped to the genome, form peaks corresponding to loci where the protein of interest was bound. (Figure from [36].)

4.2.6 Single-molecule in-situ sequencing

Previous studies comparing single-cell and single-nuclear transcription focus on technical effects (e.g. cell type biases and transcriptional stress responses) arising from sample preparation [10, 7]. However, numerous studies have identified systematic differences between the gene composition of nuclear and cytoplasmic mRNA pools, and associated these with major

biological processes [12, 13, 14, 15, 16]. It is conceivable that favoring one compartment over the other risks undersampling genes involved in such processes.

As any biological compartment specificity would be confounded with stress responses arising from tissue dissociation, differential expression in single-cell versus single-nucleus sequencing cannot tell us whether this biological bias is recapitulated in either technology – we require measurements of gene expression in both fractions in cells that undergo identical sample preparation protocols (ideally, in the exact same cells). Highly-multiplexed single-molecule imaging experiments satisfy this requirement: instantiations of this approach (such as MERFISH [37] and SeqFISH+ [38]) label a sample with hundreds to thousands of polynucleotide probes, complementary to selected genes, and sequence these in-situ via fluorescent readout of a unique barcode per probe. The process is illustrated in Figure 4.6.

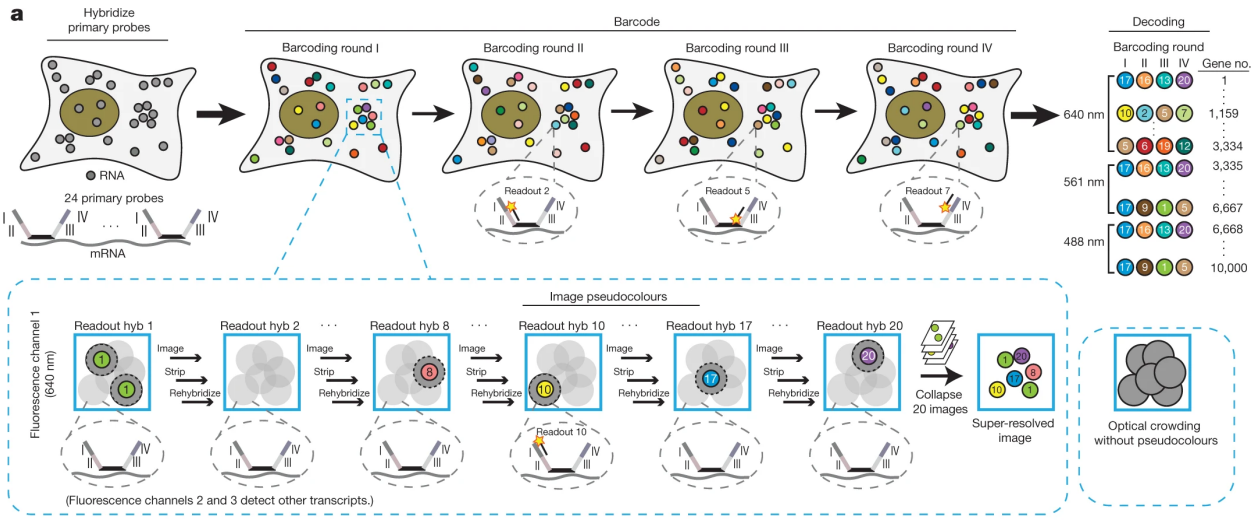

Figure 4.6: Fluorescence in-situ hybridization (FISH) optically measures single mRNA molecules in-situ using fluorescent probes complementary to a target mRNA. To avoid excessive spectral overlap or optical crowding, multiplexed methods such as SeqFISH+ use multiple probes per gene (top) and separate readout of groups genes in time (bottom). Following decoding of barcodes to genes, spot detection, and cell segmentation, the output of such an experiment is a set of 2-D coordinates per cell, each representing an individual RNA molecule of a known gene. (Figure from [38].)

In-situ sequencing datasets permit joint quantification of the cytoplasmic and nuclear mRNA fractions in single cells, which lets us directly compare gene-wise means and variances without needing to control for complex technical or biological confounders. However, such data are limited by the number and selection criteria of the probes.

4.2.7 Gene regulatory network inference

Inference of gene regulatory networks (GRNs) from omics data is a long-standing open problem in computational biology. It has followed parallel developments in sequencing technologies: while the earliest algorithms used bulk RNA sequencing data [39], today single-cell sequencing offers orders of magnitude more samples from which to learn.

Typically, GRN inference is understood to mean identifying links between transcription factors (TFs), and the genes whose expression they physically regulate, by detecting statistical dependences in gene expression data [40]. Such links can be signed, indicating activatory or inhibitory relationships. Algorithmic implementations of GRN inference have been diverse, ranging from correlation clustering [41] and penalized linear regression [42] to complex nonlinear models [23, 43, 44], and are highly dependent on the data modalities (e.g. bulk RNA-sequencing [41], single-cell RNA-sequencing [23], multimodal assays [43], RNA velocity [44]) and experimental designs (e.g. time courses [24], CRISPR perturbations [45]) to which they are applied.

Gene regulatory inference can also be viewed as an instance of a more general problem in machine learning – causal structure learning – and algorithms from that line of research have been translated to this context [46, 47]. However, the common view of GRN inference in computational biology, embodied in algorithms such as SCENIC [23], is more stringent: by requiring each edge to represent a physical binding interaction between a TF and a regulatory genomic locus, we rule out edges that correspond to indirect causal relationships between genes. (In practice, the only causal edges one can learn from exclusively gene expression data are indirect regardless: a TF does not act on a target through its mRNA transcript, but through the protein it encodes.) This constraint leads in principle to worse predictive models, but is easier to interpret and more conducive to experimental validation. It also eases the difficulty of directing causal edges: whether or not a gene encodes a transcription factor is usually known in advance. This (with the exception of TF-TF links) reduces GRN inference to sparse feature selection – 'among the set of known TFs, which are regulators of this gene?' – and we view the problem through that lens in this thesis.

4.3 Methods

4.3.1 Datasets

Standards and Technologies Working Group

Following the filtering, integration, and annotation procedures carried out by collaborators in the Mereu lab of the Josep Carreras Institute, we received a total of $\sim 130k$ cells, across 14 different types, measured by 3 sequencing technologies (single-cell RNA sequencing, single-nucleus multiome sequencing, single-cell CAGE sequencing) in 19 different biological replicates of kidney cortex. In Figure 4.7, we observe substantial variability in cell type recovery rates and read depth within the same biological samples, reflecting already-known biases of these two assays arising from dissociation protocols [7], which we later control for by resampling.

External datasets

To validate inferred TF activities, we use immunohistochemistry annotations from version 23 of the Human Protein Atlas [30]. We use TFs with a motif in JASPAR 2022 [48] that have an 'Enhanced' antibody score, and label a TF as 'active' if it is annotated as detected

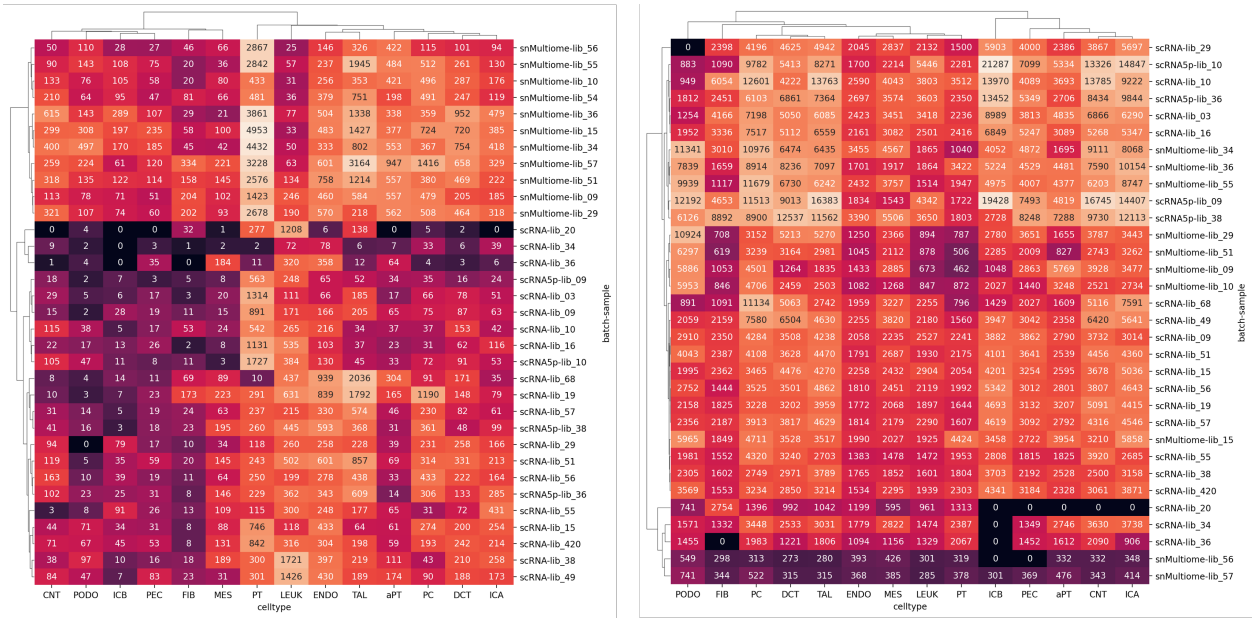

Figure 4.7: Number of cells (left) and median reads per cell (right) captured per type, sample, and assay. Cell type recovery rates cluster strongly according to method of isolation (single-cell versus single-nucleus). Certain samples also show major quality issues.

in nuclei (of any kidney cell type), and 'inactive' if it was detected only in cytoplasm or not detected at all. This yields 42 active and 102 inactive TFs.

For ground-truth TF-target links, we query ChIP-Atlas [49] for TFs in any human kidney-derived cell type passing a significance threshold of 50. To link these with target genes, we only consider peaks with ±500bp of a known transcription start site, which we obtain from version 4.1 of RefTSS [50]. Intersecting with JASPAR 2022 yields 112 TFs with promoters in 14402 genes.

To obtain sets of genes whose mRNAs specifically localize to the cytoplasm or nucleus, we use a collection of 3289 genes identified via APEX-seq [14] as differentially localized in HEK-293T cells. We consider any gene exceeding a $\log_2$-fold-change of 0.75 in any nuclear compartment (nucleus, nucleolus, lamina, nuclear pore) to be expressed in nucleus, the same criterion for any cytoplasmic compartment (cytosol, endoplasmic reticulum, ER membrane, mitochondria, mitochondrial membrane), and removed any overlapping genes. This yielded 1296 genes with nuclear localization, and 1335 genes with cytoplasmic localization.

To test hypotheses regarding overdispersion and predictive generalization, we use a Seq-FISH+ dataset that imaged 219 genes in approximately 200k cells of mouse kidney, obtained from `http://kidneyviewer.spatialgenomics.com/`. As the authors of that dataset do not assign subcellular localizations, we used Cellpose [51] to segment their provided DAPI image, and labelled any transcript overlapping a segmented nucleus as nuclear-localized. We only included cells for which more than 50 unique molecules were detected in each of the cytoplasm and nucleus.

4.3.2 Algorithms

Resampling

The acquired sequencing data exhibit substantial heterogeneity with respect to cell type coverage and read depth. As we are interested in how each assay acts as a view of biology, we elect to control for this variability, whose origins are primarily technical in origin and have been well-characterized before [10]. For each of our comparisons, we therefore carry out three sequential steps of downsampling: first, to control for donor variability, we only include biological samples that were measured in both assays under consideration. Within each sample, we next control for cell type biases across assays. Using the cell type annotations generated by our collaborators, we uniformly downsample cells such that the number of cells within each (sample, cell-type) pair is the same across assays. Finally, we control for cell-type-dependent effects in read depth by downsampling reads within each (sample, cell-type) pair to the same total reads across assays, while preserving each individual cell's fraction of the total reads, using ScanPy's [52] `sc.pp.downsample_counts` function. (Importantly, we do not balance read depth by the common approach of multiplicative rescaling – e.g. `sc.pp.normalize_total` – as this would affect our metric of dispersion.)

Gene regulatory network inference

We use least-squares linear regression with an elastic-net penalty to infer statistical links. Given N cells with expression matrices X for transcription factors and Y for targets, we minimize:

$$\min_{\beta} \frac{1}{N}\|Y - X\beta\|_F^2 + \alpha(1-\eta)\|\beta\|_F^2 + \alpha\eta\|\beta\|_{1,1} \tag{4.1}$$

where $\|\cdot\|_F^2$ denotes the squared Frobenius norm, $\|\cdot\|_{1,1}$ is the element-wise 1-norm, and $\alpha \in [0, \infty)$ and $\eta \in [0, 1]$ are regularization parameters. While more sophisticated approaches exist (e.g. [23, 43]), we select the elastic net because it is a performant, interpretable approach that straightforwardly permits incorporating epigenomic side-information when available, and has been previously used for gene regulatory inference by a major consortium [42].

We independently regress the expression of each gene against the expression of the intersection of the 720 vertebrate transcription factors listed in JASPAR 2022 with the genes observed in each assay, which we further limit to only those transcription factors supported by epigenomic (i.e. open chromatin or active promoter) evidence when available.

For speed we implemented elastic net regression on the GPU in PyTorch [53]. Our implementation also permits us to straightforwardly incorporate constraints on allowed TF-target links via multiplicative masking of the TFs-by-targets coefficient matrix β. We fit the model by running 1000 steps of (batch) gradient descent with a learning rate of 10, implementing the L1 portion of the elastic-net penalty via soft-thresholding [54]. As conditioning, we first preprocess inputs and targets by applying a $\log(1+x)$ transformation and root-mean-square-normalizing each feature. We only include input or target features detected in at least 30 cells, in every sample and assay within a given comparison. We fix the L1 ratio η to 0.5, and use 5-fold cross-validation (holding out cells within-sample) to select the regularization

parameter $\alpha \in \{5e{-}5, 1e{-}4, 2e{-}4, 3e{-}4\}$ that maximizes mean gene-wise Pearson correlation. We moreover do not fit an intercept term – we found test performance to consistently improve when omitting this parameter.

Epigenomic prior networks

With multiome and CAGE-seq, we have the option of constraining permitted regulatory network topologies to only those edges with mechanistic evidence of the transcription factor binding to loci near the target gene. We use MOODS [55] to associate genomic loci with JASPAR 2022 transcription factors via motif enrichment, filtering using the default threshold p-value 10^{-5}. It is straightforward to link these (TF, locus) pairs with target genes in CAGE-seq, as each locus is detected as a promoter of a particular gene. However, connecting ATAC-seq peaks with transcribed genes is more involved. We employ an approach similar to FigR [28], where we identify correlated pairs of peak accessibility and gene expression variables within a wide genomic window to allow for distal interactions. Specifically, we first link each epigenomic peak to any gene with a RefTSS transcription start site within ±100kb. For each gene, we then filter these links by computing the Pearson correlation between the vectors of log-transformed per-cell gene expression and peak accessibility, plus a set of 100 background peaks with similar GC content and total accessibility. We keep gene-peak links with both positive Pearson correlation and Z-score > 1.645 relative to the background correlations, as well as any gene-peak link where the peak lies in a promoter region of the gene, which we take as ±500bp of any corresponding TSS.

4.3.3 Evaluation metrics

As we lack ground-truth regulatory networks in this setting, we instead quantitatively compare gene regulatory inferences according to a number of criteria that reflect desirable qualitative properties.

1. **Within-assay predictive error**. We expect a more accurate gene regulatory network should lead to lower predictive error on held-out cells. The design of our experiment gives a natural splitting strategy: we fit a GRN to one sample, and evaluate it on each of the rest. We quantify this as the mean gene-wise Pearson correlation, across all genes and sample pairings.

2. **Within-assay network reproducibility**. We quantify reproducibility of regulatory inferences via pairwise comparisons of networks within an assay. Specifically, for each target gene, we take the size of the intersection of the sets of assigned TFs between the two networks (i.e. with nonzero elastic net coefficients) over the union, and report the average across genes and pairs of networks. We note this assumes conservation of regulatory mechanisms across different biological samples.

3. **Network sparsity**. Among regulatory models with equal predictive power, Occam's razor tells us to prefer the sparsest. This is quantified as the mean number of TFs selected per gene (in-degree), averaged over all networks.

4. **Cross-assay predictive error**. This is a stronger version of the first criterion: we now evaluate a model trained on data from one assay against expression data measured in the *other* (again on held-out biological samples). Assuming observable regulatory mechanisms are conserved across assays, and simply vary in their signal-to-noise ratio, we hypothesize an assay with lower noise will make these easier to detect, yielding models that generalize better to the noisier assay. This is again computed as the mean Pearson correlation per gene.

5. **Cross-assay network complementarity**. We wish to know the extent to which the regulatory links learnable from one assay are also present in the other. This is measured as gene-wise recall of TFs – considering a pair of networks from a 'source' and a 'target' assay, for a given gene it is the fraction of TFs in the latter that are present in the former – averaged over all genes and sample pairs.

6. **Protein expression evidence for TFs**. A necessary condition for a TF to participate in transcriptional regulation is that it physically co-localize with the events of transcription (i.e. in the nucleus). We compare assays in terms of how much regulatory influence is apportioned to TF proteins with confident nuclear localization in kidney, per the Human Protein Atlas. We specifically quantify this as the ratio of the squared 2-norm of elastic-net coefficients outbound from such TFs, to the 2-norm of coefficients for all TFs confidently detected in HPA kidney samples (irrespective of localization). A potential drawback of this approach arises if the TF is active but expressed at levels below the detection limit of IHC, or if antibody used in the HPA does not bind the TF in its active conformation – such cases would be incorrectly labelled as false positives.

7. **ChIP-seq binding evidence in promoters**. We assume that a high-confidence peak observed in a bulk ChIP-seq experiment of a particular TF, found in the promoter region of a gene, implies the TF is a ground-truth regulator of that gene. However this only provides a subset of ground-truth edges: in particular it omits interactions with enhancer regions, which we cannot reliably associate to a target gene using ChIP-seq alone. We therefore only evaluate inferred edges against ChIP-seq data at the level of recall: specifically, for each TF that was assayed in kidney-derived cells in ChIP-Atlas, we report the size of the intersection between the set of genes to whose promoter that TF bound, and the set of inferred regulatory targets; divided by the size of the former. We point out this risks overestimating false negatives if a TF that binds in some kidney-derived cell line does not do so in adult kidney.

4.4 Results and Discussion

4.4.1 Single-cell versus single-nuclear RNA sequencing

We first investigate differences between gene regulatory networks derived from single-cell versus single-nuclear RNA sequencing, which we attempt to explain in terms of statistical differences between cytoplasmic and nuclear transcript pools.

Datasets employed

Among the kidney samples acquired by the STWG, 10 were subjected to both single-cell RNA sequencing and single-nuclear multiome sequencing. Among these, we discard 4 samples displaying unusually low sampling of at least one cell type, leaving the samples `lib_09`, `lib_10`, `lib_15`, `lib_29`, `lib_51`, and `lib_55`. Following subsampling, each sample measures an average of 1878 cells, with 2728 reads each.

Differential localization

In Figure 4.8, we first investigate whether gene-specific biases for cytoplasmic versus nuclear transcript localization influence mean expression in either assay.

As differences between single-cell and single-nucleus expression reflect differences in sample preparation [7], we cannot directly identify differentially localized transcripts simply as those differentially expressed between the two assays. Instead, we use genesets identified by APEX-seq in HEK-293T cells as having preferential nuclear (N=1137) or cytoplasmic (N=1237) mRNA localization. We compare the expression of these genes within assays to a background set of 2994 genes also measured in APEX-seq.

For single-cell sequencing, we see this is the case: cytoplasmic transcripts are oversampled (p = 1e-6, two-tailed T-test), and nuclear transcripts are undersampled (p = 0.01) in single-cell sequencing. However, no significant differential expression is observed in single-nucleus sequencing for either compartment. We speculate that compartmental specificity in quiescent cells might therefore arise from cytoplasmic degradation rates, with nuclear export rates – although known to vary widely in immune cells [18] – playing less of a role.

We also confirm nuclear and cytoplasmic genes enrich for differential expression across assays, finding log-fold-change of single-nucleus versus single-cell is significantly more positive for nuclear-localized genes (p=0.03) than background genes, and significantly more negative (p=6e-8) for cytoplasm-localized genes.

Overdispersion

Battich et al [21] show cytoplasmic transcript counts have lower dispersion than nuclear transcripts: transcriptional bursting noise is buffered by the pool of accumulated mRNA in the nucleus, which is exported into the cytoplasm at slower timescales than bursting. Having observed an association between compartment localization and measured mean gene expression, we next hypothesized these differences in biological noise levels between compartments would also appear in single-cell and single-nucleus sequencing of a complex tissue.

We quantify noise by the index of dispersion [56], which is the observed variance normalized to that expected of a Poisson process ($d = 1$):

$$d_X = \frac{s_X^2}{\bar{x}}$$

where s_X^2 and $\bar{x}$ are the sample variance and mean of random variable X respectively. We compute two indices of dispersion for each gene, by counting mRNA molecules in separately the nuclear and the cytoplasmic fraction, and compare them via paired T-test.

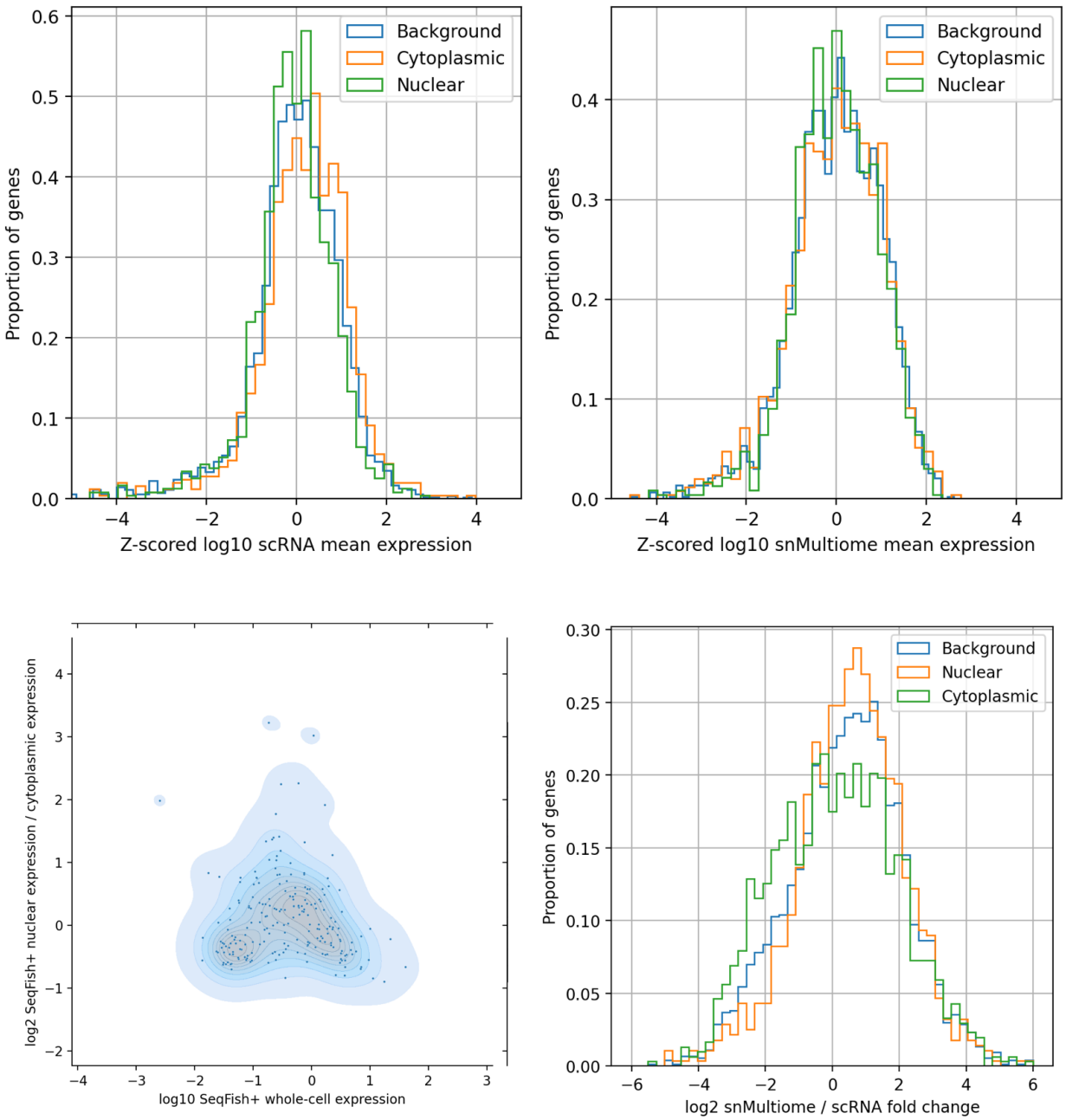

Figure 4.8: Expression of compartment-specific genes identified by APEX-seq in single-cell (top left) and single-nucleus (top right) sequencing. Bottom left: compartment specificity is uncorrelated (r=-0.05, p=0.26) with overall gene expression in SeqFISH+. Bottom right: compartment-specific genes exhibit expected differential expression between assays.

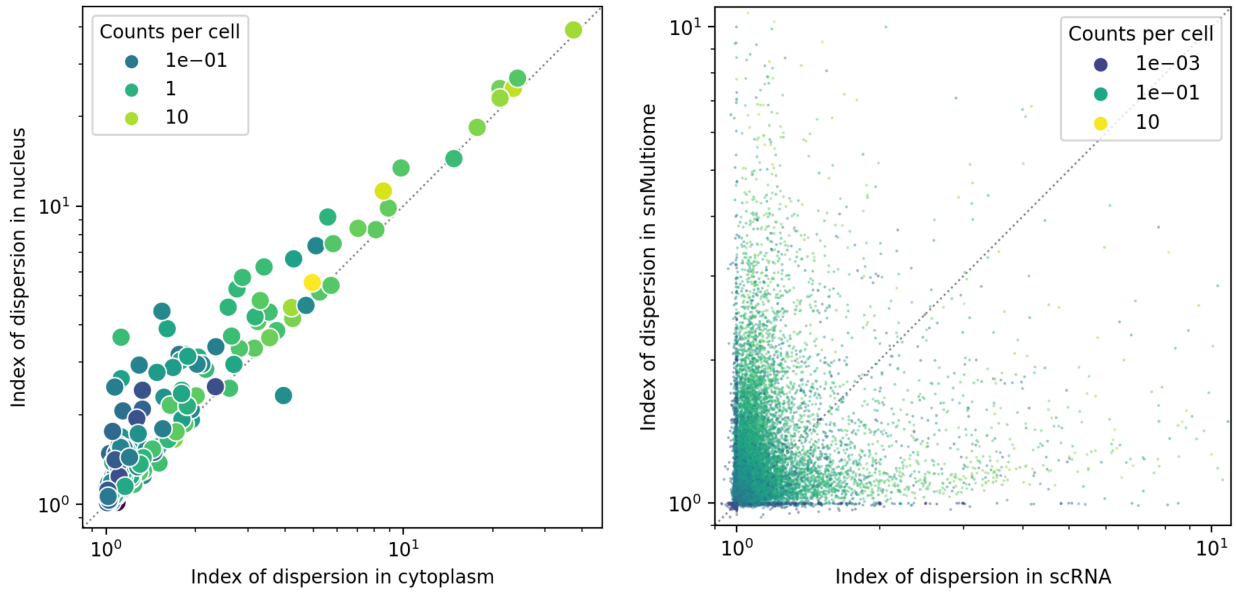

Figure 4.9: Left: SeqFISH+ detects overdispersion in nuclear transcript counts relative to cytoplasmic transcript counts of 219 genes. Right: 15,306 genes also show greater dispersion in single-nucleus versus single-cell sequencing.

As shown in Figure 4.9, we confirm our hypothesis: we observe significant (p = 9e-7, paired two-tailed T-test) over-dispersion of the same gene in single-nucleus sequencing relative to single-cell. To verify this is a biological effect in kidney, and not solely an artifact of tissue dissociation, we also compare cytoplasmic and nuclear indices of dispersion of 219 genes imaged in mouse kidney via SeqFISH+, and observe significant overdispersion there as well (p = 2e-13).

Gene regulatory inference

Having established prior reason to expect differences in nuclear versus whole-cell (i.e. primarily cytoplasmic) inferences, and controlled for cell-type biases and read depth, we now compare single-cell versus single-nucleus sequencing in terms of predictive power, reproducibility, and biological integrity of inferred gene regulatory networks. We fit those networks using 142 transcription factors and 4272 target genes that pass our quality filtering, and evaluate them using the aforementioned metrics.

We show our results in Table 4.1. While performance in absolute terms is overall poor, this is to be expected: modelling gene expression as solely a linear function of the mRNA expression of a small set of transcription factors is a poor approximation. It nonetheless reveals significant differences between the two assays under investigation: notably, single-nucleus expression is easier to predict (with our model of choice) than single-cell expression. We speculate unmeasured or non-transcriptional causes of mRNA abundance (e.g. genes involved in mRNA degradation) might play a greater role in the cytoplasm than the nucleus. The greater density of networks inferred from single-cell data (such causes might correlate with linear combinations of several TFs), superior reproducibility within-assay of

	Single-cell RNA-seq	Single-nucleus RNA-seq
Within-assay Pearson r	0.288 ± 0.004	0.319 ± 0.004
Within-assay edge IoU	0.257 ± 0.003	0.306 ± 0.003
Cross-assay Pearson r	0.276 ± 0.003	0.233 ± 0.003
Cross-assay edge recall	0.275 ± 0.003	0.18 ± 0.002
Gene in-degree	23.574 ± 0.192	15.536 ± 0.153
IHC TF weight	0.663 ± 0.006	0.555 ± 0.008
ChIP-seq edge recall	0.158 ± 0.05	0.073 ± 0.031

Table 4.1: Performance metrics for regulatory networks inferred from single-cell versus single-nucleus RNA sequencing. 95% confidence intervals of the mean indicated.

single-nucleus (said combinations might vary across samples) and far poorer out-of-assay performance of single-nucleus networks relative to single-cell would also support this.

We also considered whether superior predictability of nuclear versus whole-cell mRNA abundances might relate to transcriptional bursting. If transcriptional noise is correlated across different targets of the same transcription factor, it is conceivable this correlated noise would be useful for detecting co-regulated genes, and moreover attenuated in the cytoplasmic mRNA pool by nuclear buffering. However, we do not believe this is the correct explanation here, as we fit an independent model to each gene, and only share information between potentially co-regulated genes very weakly (during hyperparameter selection).

Surprisingly, despite its poorer predictive performance, scRNA apportions more signal to TFs with proteomic evidence than snRNA does, and recalls far more ChIP-seq edges. The reason for this is unclear to us, but may relate to the cytoplasmic mRNA abundance of a TF being a closer proxy in principle to its protein expression than its nuclear mRNA abundance.

To test whether these findings arise from biological effects, as opposed to technical biases between single-cell and single-nucleus sequencing, we investigated whether these findings were reproducible in SeqFISH+ data of mouse kidney. This evaluation faces limitations: of the 219 genes probed in this dataset, only 8 represent TFs in JASPAR 2022. We also use 5-fold cross validation within a single biological sample instead of evaluation on a separate sample. However, because we have $\sim 100\times$ more cells in this scenario, we can use a more expressive model than linear regression: we choose `scikit-learn`'s [?]`HistGradientBoostingRegressor` (with default parameters). Moreover, we directly observe cytoplasmic expression in this context, as opposed to the entire cell. Table 4.2 corroborates our findings: nuclear expression is easier to predict than cytoplasmic in this context as well.

	Cytoplasm	Nucleus
Within-compartment Pearson r	0.081 ± 0.005	0.131 ± 0.007
Cross-compartment Pearson r	0.112 ± 0.007	0.072 ± 0.004

Table 4.2: Performance metrics for regulatory networks inferred from cytoplasmic versus nuclear mRNA pools in SeqFISH+ of mouse kidney.

4.4.2 Single-cell ATAC-seq vs single-cell CAGE-seq

Both single-nucleus multiome sequencing and single-cell CAGE sequencing provide joint single-cell measurements of transcription and an epigenomic modality. ATAC-seq samples open chromatin across the entire genome, permitting unbiased detection of different classes of genomic regions; but is very sparse and only indirectly permits connecting genomic loci to genes through proximity and correlation. CAGE-seq only measures active promoter sequences, but directly associates each measured transcript with one such sequence. Moreover, this association is *causal*: observing a promoter sequence at the 5' end of a transcript implies the binding of a TF to that specific site in the genome was part of the transcriptional event that produced it. As it is not clear a-priori which should lead to better inferred regulatory networks, we investigate both empirically.

Datasets employed

Of the available samples, we use the three – `lib_09`, `lib_10`, `lib_36` – that were subjected to both single-nucleus multiome sequencing and single-cell CAGE sequencing. Following subsampling, each sample measures an average of 3716 cells, with 2631 reads each.

Comparing detections of TF binding sites

The epigenomic modalities of our two multimodal assays under consideration have very different sampling biases. To understand whether these could influence estimates of TF binding site accessibility, for each TF we count the overall number of reads assigned to loci (peaks or promoters) with a motif match in JASPAR 2022. We show this in Figure 4.10.

Despite greater sparsity of epigenomic data relative to transcriptomics, the greater breadth of loci sampled by ATAC-seq leads to $\sim 2\times$ greater reads detected overall. There is a strong correlation (r=0.87) between the number of reads detected per TF in the two assays, which becomes almost perfect (r=0.98) when restricting to loci overlapping a known TSS in RefTSS. However this may reflect prior multiplicity of motifs in the genome, rather than strong correlation between the measurements. The near-unity slope between the two assays for TSS loci was unexpected and might be coincidental or an artifact of subsampling.

We also examined the redundancy in TF activity information between loci overlapping known TSS sites, and non-overlapping loci, within each assay. Specifically we assign, to each TF with gene expression evidence, the total count of epigenomic reads across all cells within loci matching that TF, for either class of locus. As expected, most reads in ATAC-seq lie outside promoters (which should only constitute a small fraction of the accessible genome) while CAGE-seq is biased toward promoters by design (but still detects transcribed 5'-ends that do not match any known TSS). In both assays these are strongly correlated (p=0.63 in ATAC-seq, p=0.70 in CAGE-seq), which may also arise from prior motif multiplicity.

Comparing estimates of promoter multiplicity

A major appeal of CAGE-seq is its ability to detect and quantify alternate promoter usage. As shown in Figure 4.11, we investigated whether ATAC-seq is also capable of detecting alternate promoter usage when it occurs.

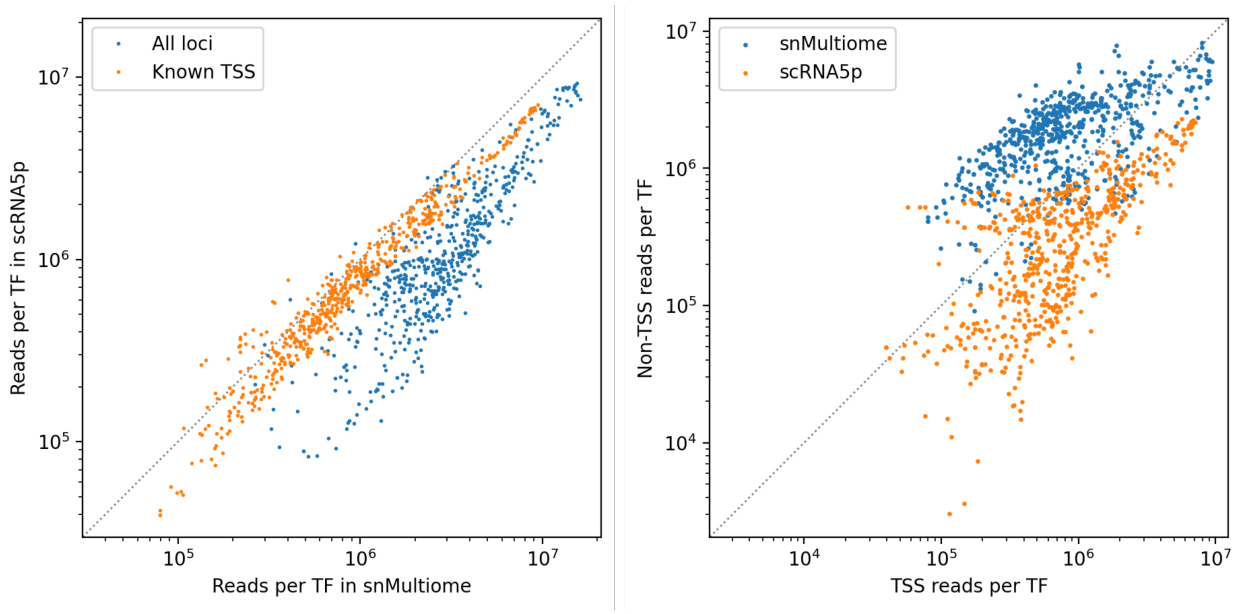

Figure 4.10: Left: cross-assay comparison of total reads in loci matched to JASPAR 2022 motifs, overall (blue) and restricted to those overlapping known promoters in RefTSS (orange); right: within-assay comparison of total reads in loci overlapping versus outside of known promoter regions.

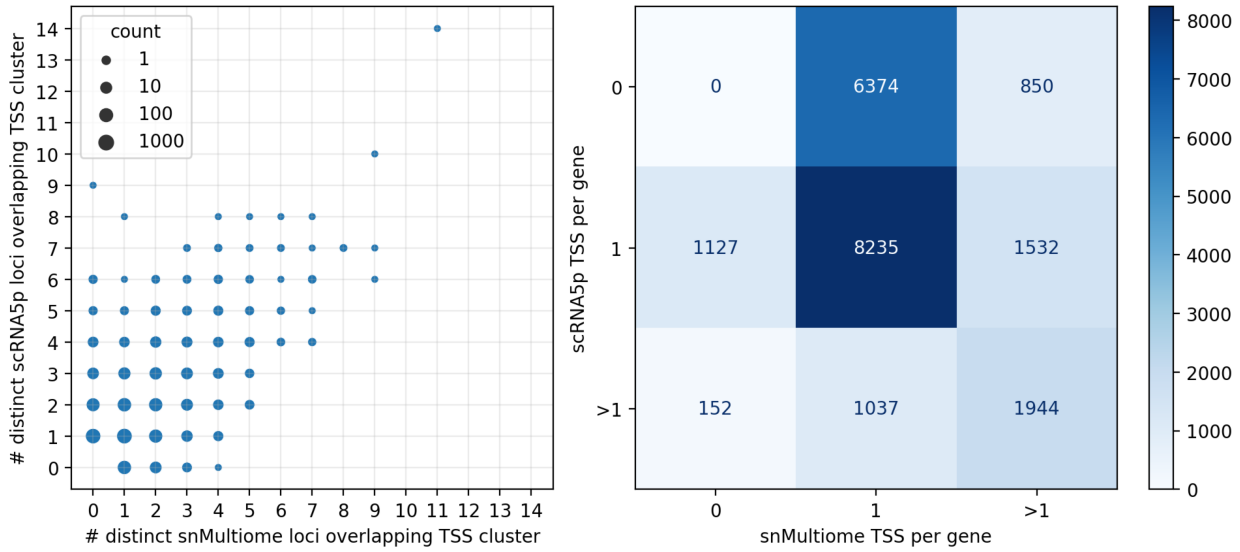

Figure 4.11: Left: for each epigenomic modality, we count the number of unique detected loci (promoters / peaks) overlapping any known TSS; right: we look at either assay as a classifier of alternate promoter usage (not detected, single promoter, multiple promoter)

For each gene with expression evidence and associated promoter cluster in RefTSS, we count the number of distinct peaks overlapping that cluster, and compare to the number of (known) promoters identified by CAGE-seq. We observe a moderate (r=0.39) positive correlation between the number of active promoters per gene detected by either assay. We

can also consider each assay as a classifier of promoter usage (not detected, single, multiple). ATAC-seq greatly overestimates promoter multiplicity in this case, which is unsurprising: calling a peak an active promoter, simply because it overlaps a possible TSS of an expressed gene, is less precise than the direct transcriptional readout from CAGE-seq. While it is conceivable sequence features could be used to further refine this, we leave development of such a classifier as a direction for future work.

Gene regulatory network inference

Our ultimate objective in this section is comparison of ATAC-seq versus CAGE-seq as the epigenomic modality of a multimodal measurement. Before this comparison, we first carry out an ablation of the epigenomic modality, studying its effect on GRN inference relative to the respective transcriptomic modality of each assay alone. We might expect a-priori that by excluding TF-target links that do not represent a direct physical regulatory interaction (i.e. spurious correlations), using the epigenomic modality to constrain the set of regulatory links learnable from transcriptomics data will lead to predictive models that generalize better. However, this also limits expressiveness of the models that can be learned, and will moreover lead to worse predictions if the 'spurious' correlations actually reflect confounders whose activity is conserved across samples.

	Single-nucleus multiome	**Single-nucleus RNA-seq**
Pearson r	0.359 ± 0.004	0.375 ± 0.004
Within-assay edge IoU	0.281 ± 0.004	0.303 ± 0.004
Cross-assay edge recall	0.246 ± 0.004	0.317 ± 0.004
Gene in-degree	11.113 ± 0.125	14.666 ± 0.155
IHC TF weight	0.709 ± 0.010	0.592 ± 0.008
ChIP-seq edge recall	0.086 ± 0.040	0.064 ± 0.039

Table 4.3: Performance metrics for single-nucleus multiome, with and without ATAC-seq constraints. (We only list a single Pearson r in the ablations as the transcriptomics datasets compared across 'assays' are the same.)

We first compare the effect of including ATAC-seq as a prior constraint on GRNs inferred from single-nucleus multiome sequencing; fitting (here and in all subsequent comparisons) 143 TFs against 4420 targets. The results are shown in Table 4.3. As expected, this prior selects more biologically plausible networks: they assign more weight to TFs with nuclear protein expression and recover more known TF-promoter links from ChIP-seq. Surprisingly, despite greater biological plausibility, the constraint hurts predictive performance: this could reflect true TF-target links (or links necessary to capture non-TF confounders) being excluded by insufficiently sensitive ATAC-seq or motif enrichment. The poorer within-assay reproducibility of constrained networks may reflect variability across samples of the ATAC-seq prior itself, which necessitates its own inference procedure.

In Table 4.4 we carry out the same ablation with single-cell CAGE sequencing, where our epigenomic constraint consists of restricting allowed TF-target links to those where the TF was detected in an active promoter. As expected this yields much sparser networks –

Metrics	Single-cell CAGE-seq	Single-cell 5'-end RNA-seq
Pearson r	0.311 ± 0.003	0.354 ± 0.003
Within-assay edge IoU	0.409 ± 0.005	0.212 ± 0.003
Cross-assay edge recall	0.121 ± 0.002	0.236 ± 0.003
Gene in-degree	10.819 ± 0.126	22.002 ± 0.179
IHC TF weight	0.750 ± 0.010	0.540 ± 0.008
ChIP-seq edge recall	0.142 ± 0.045	0.140 ± 0.050

Table 4.4: Performance metrics for single-cell CAGE-seq, with and without active promoter constraints.

potentially reflecting exclusion of TF-target interactions that bind to enhancer regions – and comes (as with multiome) at a significant cost in predictive performance. However, the constraint also greatly improves accuracy of active TF selection: although it surprisingly has no effect on recall of TF-promoter links.

	Single-cell CAGE-seq	Single-nucleus multiome
Within-assay Pearson r	0.311 ± 0.003	0.359 ± 0.004
Within-assay edge IoU	0.409 ± 0.005	0.281 ± 0.004
Cross-assay Pearson r	0.312 ± 0.004	0.298 ± 0.003
Cross-assay edge recall	0.300 ± 0.004	0.296 ± 0.004
Gene in-degree	10.819 ± 0.126	11.113 ± 0.125
IHC TF weight	0.750 ± 0.010	0.709 ± 0.010
ChIP-seq edge recall	0.142 ± 0.045	0.086 ± 0.040

Table 4.5: Performance metrics for single-cell CAGE-seq versus single-nucleus multiome sequencing.

In Table 4.5, we finally compare GRN inferences from the two multimodal assays. Single-nucleus multiome sequencing yields considerably better out-of-sample predictions within-assay. However, the cross-assay predictive power is better for CAGE-seq. The more restrictive CAGE-seq prior also leads to better edge reproducibility within-assay; surprisingly, despite this restriction the cross-assay edge recall is essentially the same, and moreover CAGE-Seq attains significantly greater biological plausibility than multiome. We find this result counter-intuitive: perhaps the benefit of directly measuring causal genomic loci of transcripts outweighs the cost of excluding regulation at enhancers. We nonetheless hesitate to claim that CAGE-seq dominates ATAC-seq as an epigenomic modality for regulatory network inference: as we previously established, single-cell sequencing alone leads to better biological plausibility than single-nucleus in the absence of epigenomic constraints, and could be playing a role here as well.

4.5 Conclusions

In this analysis, we find statistical differences between cytoplasmic and nuclear mRNA pools identified in cell cultures also appear in single-cell and single-nucleus measurements in a

complex tissue sample. We find gene regulation in single-nucleus RNA expression is better modelled by a sparse linear model from known TFs to genes than it is in single-cell RNA expression – however, regulatory networks inferred from single-cell expression are more biologically plausible. Comparing epigenomic modalities, we find single-cell ATAC-seq is capable of detecting peaks corresponding to known alternate promoters, but CAGE-seq indicates accessibility does not imply activity. Constraining regulatory inferences via epigenomic information enhances biological plausibility of inferred networks in both cases; however, both assays experience reduced out-of-sample predictive power due to this integration, indicating a trade-off between model robustness and expressiveness. We additionally note our conclusions may be limited by small sample sizes, and our corresponding choice to use linear models: this was a result of controlling for upstream discrepancies in sample availability and experimental protocols.

References

[1] Abhijeet Rajendra Sonawane et al. "Understanding Tissue-Specific Gene Regulation". In: *Cell Reports* 21.4 (Oct. 2017), pp. 1077–1088. DOI: 10.1016/j.celrep.2017.10.001. URL: https://doi.org/10.1016/j.celrep.2017.10.001.

[2] Aviv Regev et al. "The Human Cell Atlas". In: *eLife* 6 (Dec. 2017). DOI: 10.7554/elife.27041. URL: https://doi.org/10.7554/elife.27041.

[3] Ashraful Haque et al. "A practical guide to single-cell RNA-sequencing for biomedical research and clinical applications". In: *Genome Medicine* 9.1 (Aug. 2017). DOI: 10.1186/s13073-017-0467-4. URL: https://doi.org/10.1186/s13073-017-0467-4.

[4] Jason D. Buenrostro et al. "Single-cell chromatin accessibility reveals principles of regulatory variation". In: *Nature* 523.7561 (June 2015), pp. 486–490. DOI: 10.1038/nature14590. URL: https://doi.org/10.1038/nature14590.

[5] Marlon Stoeckius et al. "Simultaneous epitope and transcriptome measurement in single cells". In: *Nature Methods* 14.9 (July 2017), pp. 865–868. DOI: 10.1038/nmeth.4380. URL: https://doi.org/10.1038/nmeth.4380.

[6] Akshay Iyer, Anouk A. J. Hamers, and Asha B. Pillai. "CyTOF® for the Masses". In: *Frontiers in Immunology* 13 (Apr. 2022). ISSN: 1664-3224. DOI: 10.3389/fimmu.2022.815828. URL: http://dx.doi.org/10.3389/fimmu.2022.815828.

[7] Jiarui Ding et al. "Systematic comparison of single-cell and single-nucleus RNA-sequencing methods". In: *Nature Biotechnology* 38.6 (Apr. 2020), pp. 737–746. ISSN: 1546-1696. DOI: 10.1038/s41587-020-0465-8. URL: http://dx.doi.org/10.1038/s41587-020-0465-8.

[8] Tsukasa Kouno et al. "C1 CAGE detects transcription start sites and enhancer activity at single-cell resolution". In: *Nature Communications* 10.1 (Jan. 2019). ISSN: 2041-1723. DOI: 10.1038/s41467-018-08126-5. URL: http://dx.doi.org/10.1038/s41467-018-08126-5.

[9] Simone Picelli et al. "Full-length RNA-seq from single cells using Smart-seq2". In: *Nature Protocols* 9.1 (Jan. 2014), pp. 171–181. ISSN: 1750-2799. DOI: 10.1038/nprot.2014.006. URL: http://dx.doi.org/10.1038/nprot.2014.006.

[10] Haojia Wu et al. "Advantages of Single-Nucleus over Single-Cell RNA Sequencing of Adult Kidney: Rare Cell Types and Novel Cell States Revealed in Fibrosis". In: *Journal of the American Society of Nephrology* 30.1 (Dec. 2018), pp. 23–32. ISSN: 1533-3450. DOI: 10.1681/asn.2018090912. URL: http://dx.doi.org/10.1681/ASN.2018090912.

[11] Hideya Kawaji et al. "Comparison of CAGE and RNA-seq transcriptome profiling using clonally amplified and single-molecule next-generation sequencing". In: *Genome Research* 24.4 (Mar. 2014), pp. 708–717. ISSN: 1088-9051. DOI: 10.1101/gr.156232.113. URL: http://dx.doi.org/10.1101/gr.156232.113.

[12] Beata Werne Solnestam et al. "Comparison of total and cytoplasmic mRNA reveals global regulation by nuclear retention and miRNAs". In: *BMC Genomics* 13.1 (Oct. 2012). ISSN: 1471-2164. DOI: 10.1186/1471-2164-13-574. URL: http://dx.doi.org/10.1186/1471-2164-13-574.

[13] Chenglong Xia et al. "Spatial transcriptome profiling by MERFISH reveals subcellular RNA compartmentalization and cell cycle-dependent gene expression". In: *Proceedings of the National Academy of Sciences* 116.39 (Sept. 2019), pp. 19490–19499. ISSN: 1091-6490. DOI: 10.1073/pnas.1912459116. URL: http://dx.doi.org/10.1073/pnas.1912459116.

[14] Furqan M. Fazal et al. "Atlas of Subcellular RNA Localization Revealed by APEX-Seq". In: *Cell* 178.2 (July 2019), 473–490.e26. ISSN: 0092-8674. DOI: 10.1016/j.cell.2019.05.027. URL: http://dx.doi.org/10.1016/j.cell.2019.05.027.

[15] Malgorzata Kloc, N.Ruth Zearfoss, and Laurence D. Ekin. "Mechanisms of Subcellular mRNA Localization". In: *Cell* 108.4 (Feb. 2002), pp. 533–544. ISSN: 0092-8674. DOI: 10.1016/s0092-8674(02)00651-7. URL: http://dx.doi.org/10.1016/s0092-8674(02)00651-7.

[16] Roger A Barthelson et al. "Comparison of the contributions of the nuclear and cytoplasmic compartments to global gene expression in human cells". In: *BMC Genomics* 8.1 (2007), p. 340. ISSN: 1471-2164. DOI: 10.1186/1471-2164-8-340. URL: http://dx.doi.org/10.1186/1471-2164-8-340.

[17] Keren Bahar-Halpern et al. "Nuclear Retention of mRNA in Mammalian Tissues". In: *Cell Reports* 13.12 (Dec. 2015), pp. 2653–2662. ISSN: 2211-1247. DOI: 10.1016/j.celrep.2015.11.036. URL: http://dx.doi.org/10.1016/j.celrep.2015.11.036.

[18] Diane Lefaudeux et al. "Kinetics of mRNA nuclear export regulate innate immune response gene expression". In: *Nature Communications* 13.1 (Nov. 2022). ISSN: 2041-1723. DOI: 10.1038/s41467-022-34635-5. URL: http://dx.doi.org/10.1038/s41467-022-34635-5.

[19] Maria Dermit et al. "Subcellular mRNA Localization Regulates Ribosome Biogenesis in Migrating Cells". In: *Developmental Cell* 55.3 (Nov. 2020), 298–313.e10. ISSN: 1534-5807. DOI: 10.1016/j.devcel.2020.10.006. URL: http://dx.doi.org/10.1016/j.devcel.2020.10.006.

[20] Bjorn Schwanhausser et al. "Global quantification of mammalian gene expression control". In: *Nature* 473.7347 (May 2011), pp. 337–342. ISSN: 1476-4687. DOI: 10.1038/nature10098. URL: http://dx.doi.org/10.1038/nature10098.

[21] Nico Battich, Thomas Stoeger, and Lucas Pelkmans. "Control of Transcript Variability in Single Mammalian Cells". In: *Cell* 163.7 (Dec. 2015), pp. 1596–1610. ISSN: 0092-8674. DOI: 10.1016/j.cell.2015.11.018. URL: http://dx.doi.org/10.1016/j.cell.2015.11.018.

[22] George Orphanides and Danny Reinberg. "A Unified Theory of Gene Expression". In: *Cell* 108.4 (Feb. 2002), pp. 439–451. ISSN: 0092-8674. DOI: 10.1016/s0092-8674(02)00655-4. URL: http://dx.doi.org/10.1016/s0092-8674(02)00655-4.

[23] Sara Aibar et al. "SCENIC: single-cell regulatory network inference and clustering". In: *Nature Methods* 14.11 (Oct. 2017), pp. 1083–1086. DOI: 10.1038/nmeth.4463. URL: https://doi.org/10.1038/nmeth.4463.

[24] Aditya Pratapa et al. "Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data". In: *Nature Methods* 17.2 (Jan. 2020), pp. 147–154. DOI: 10.1038/s41592-019-0690-6. URL: https://doi.org/10.1038/s41592-019-0690-6.

[25] Yuhan Hao et al. "Integrated analysis of multimodal single-cell data". In: *Cell* 184.13 (June 2021), 3573–3587.e29. DOI: 10.1016/j.cell.2021.04.048. URL: https://doi.org/10.1016/j.cell.2021.04.048.

[26] Fiorella C. Grandi et al. "Chromatin accessibility profiling by ATAC-seq". In: *Nature Protocols* 17.6 (Apr. 2022), pp. 1518–1552. ISSN: 1750-2799. DOI: 10.1038/s41596-022-00692-9. URL: http://dx.doi.org/10.1038/s41596-022-00692-9.

[27] Alicia N Schep et al. "chromVAR: inferring transcription-factor-associated accessibility from single-cell epigenomic data". In: *Nature Methods* 14.10 (Aug. 2017), pp. 975–978. ISSN: 1548-7105. DOI: 10.1038/nmeth.4401. URL: http://dx.doi.org/10.1038/nmeth.4401.

[28] Vinay K. Kartha et al. "Functional inference of gene regulation using single-cell multi-omics". In: *Cell Genomics* 2.9 (Sept. 2022), p. 100166. ISSN: 2666-979X. DOI: 10.1016/j.xgen.2022.100166. URL: http://dx.doi.org/10.1016/j.xgen.2022.100166.

[29] Kouichi Kimura et al. "Diversification of transcriptional modulation: Large-scale identification and characterization of putative alternative promoters of human genes". In: *Genome Research* 16.1 (Dec. 2005), pp. 55–65. ISSN: 1088-9051. DOI: 10.1101/gr.4039406. URL: http://dx.doi.org/10.1101/gr.4039406.

[30] Peter J. Thul and Cecilia Lindskog. "The human protein atlas: A spatial map of the human proteome". In: *Protein Science* 27.1 (Oct. 2017), pp. 233–244. ISSN: 1469-896X. DOI: 10.1002/pro.3307. URL: http://dx.doi.org/10.1002/pro.3307.

[31] Wei Chang Colin Tan et al. "Overview of multiplex immunohistochemistry/immunofluorescence techniques in the era of cancer immunotherapy". In: *Cancer Communications* 40.4 (Apr. 2020), pp. 135–153. ISSN: 2523-3548. DOI: 10.1002/cac2.12023. URL: http://dx.doi.org/10.1002/cac2.12023.

[32] SyedAhmed Taqi et al. "A review of artifacts in histopathology". In: *Journal of Oral and Maxillofacial Pathology* 22.2 (2018), p. 279. ISSN: 0973-029X. DOI: 10.4103/jomfp.jomfp_125_15. URL: http://dx.doi.org/10.4103/jomfp.JOMFP_125_15.

[33] José A. Ramos-Vara and Luke B. Borst. "Immunohistochemistry". In: *Tumors in Domestic Animals* (Nov. 2016), pp. 44–87. DOI: 10.1002/9781119181200.ch3. URL: http://dx.doi.org/10.1002/9781119181200.ch3.

[34] Anna Asplund et al. "Antibodies for profiling the human proteome-The Human Protein Atlas as a resource for cancer research". In: *PROTEOMICS* 12.13 (July 2012), pp. 2067–2077. ISSN: 1615-9853. DOI: 10.1002/pmic.201100504. URL: http://dx.doi.org/10.1002/pmic.201100504.

[35] Peter J. Park. "ChIP–seq: advantages and challenges of a maturing technology". In: *Nature Reviews Genetics* 10.10 (Sept. 2009), pp. 669–680. ISSN: 1471-0064. DOI: 10.1038/nrg2641. URL: http://dx.doi.org/10.1038/nrg2641.

[36] Krassimira Botcheva et al. "Distinct p53 genomic binding patterns in normal and cancer-derived human cells". In: *Cell Cycle* 10.24 (Dec. 2011), pp. 4237–4249. ISSN: 1551-4005. DOI: 10.4161/cc.10.24.18383. URL: http://dx.doi.org/10.4161/cc.10.24.18383.

[37] Kok Hao Chen et al. "Spatially resolved, highly multiplexed RNA profiling in single cells". In: *Science* 348.6233 (Apr. 2015). ISSN: 1095-9203. DOI: 10.1126/science.aaa6090. URL: http://dx.doi.org/10.1126/science.aaa6090.

[38] Chee-Huat Linus Eng et al. "Transcriptome-scale super-resolved imaging in tissues by RNA seqFISH+". In: *Nature* 568.7751 (Mar. 2019), pp. 235–239. ISSN: 1476-4687. DOI: 10.1038/s41586-019-1049-y. URL: http://dx.doi.org/10.1038/s41586-019-1049-y.

[39] Nir Friedman et al. "Using Bayesian Networks to Analyze Expression Data". In: *Journal of Computational Biology* 7.3–4 (Aug. 2000), pp. 601–620. ISSN: 1557-8666. DOI: 10.1089/106652700750050961. URL: http://dx.doi.org/10.1089/106652700750050961.

[40] Kyle Akers and T.M. Murali. "Gene regulatory network inference in single-cell biology". In: *Current Opinion in Systems Biology* 26 (June 2021), pp. 87–97. ISSN: 2452-3100. DOI: 10.1016/j.coisb.2021.04.007. URL: http://dx.doi.org/10.1016/j.coisb.2021.04.007.

[41] Peter Langfelder and Steve Horvath. "WGCNA: an R package for weighted correlation network analysis". In: *BMC Bioinformatics* 9.1 (Dec. 2008). ISSN: 1471-2105. DOI: 10.1186/1471-2105-9-559. URL: http://dx.doi.org/10.1186/1471-2105-9-559.

[42] Daifeng Wang et al. "Comprehensive functional genomic resource and integrative model for the human brain". In: *Science* 362.6420 (Dec. 2018). ISSN: 1095-9203. DOI: 10.1126/science.aat8464. URL: http://dx.doi.org/10.1126/science.aat8464.

[43] Carmen Bravo Gonzalez-Blas et al. "SCENIC+: single-cell multiomic inference of enhancers and gene regulatory networks". In: *Nature Methods* 20.9 (July 2023), pp. 1355–1367. ISSN: 1548-7105. DOI: 10.1038/s41592-023-01938-4. URL: http://dx.doi.org/10.1038/s41592-023-01938-4.

[44] Xiaojie Qiu et al. "Mapping transcriptomic vector fields of single cells". In: *Cell* 185.4 (Feb. 2022), 690–711.e45. ISSN: 0092-8674. DOI: 10.1016/j.cell.2021.12.045. URL: http://dx.doi.org/10.1016/j.cell.2021.12.045.

[45] Joseph M. Replogle et al. "Mapping information-rich genotype-phenotype landscapes with genome-scale Perturb-seq". In: *Cell* 185.14 (July 2022), 2559–2575.e28. ISSN: 0092-8674. DOI: 10.1016/j.cell.2022.05.013. URL: http://dx.doi.org/10.1016/j.cell.2022.05.013.

[46] Nicolai Meinshausen et al. "Methods for causal inference from gene perturbation experiments and validation". In: *Proceedings of the National Academy of Sciences* 113.27 (July 2016), pp. 7361–7368. ISSN: 1091-6490. DOI: 10.1073/pnas.1510493113. URL: http://dx.doi.org/10.1073/pnas.1510493113.

[47] Yuhao Wang et al. "Permutation-based Causal Inference Algorithms with Interventions". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/275d7fb2fd45098ad5c3ece2ed4a2824-Paper.pdf.

[48] Jaime A Castro-Mondragon et al. "JASPAR 2022: the 9th release of the open-access database of transcription factor binding profiles". In: *Nucleic Acids Research* 50.D1 (Nov. 2021), pp. D165–D173. ISSN: 1362-4962. DOI: 10.1093/nar/gkab1113. URL: http://dx.doi.org/10.1093/nar/gkab1113.

[49] Zhaonan Zou et al. "ChIP-Atlas 2021 update: a data-mining suite for exploring epigenomic landscapes by fully integrating ChIP-seq, ATAC-seq and Bisulfite-seq data". In: *Nucleic Acids Research* 50.W1 (Mar. 2022), W175–W182. ISSN: 1362-4962. DOI: 10.1093/nar/gkac199. URL: http://dx.doi.org/10.1093/nar/gkac199.

[50] Imad Abugessaisa et al. "refTSS: A Reference Data Set for Human and Mouse Transcription Start Sites". In: *Journal of Molecular Biology* 431.13 (June 2019), pp. 2407–2422. ISSN: 0022-2836. DOI: 10.1016/j.jmb.2019.04.045. URL: http://dx.doi.org/10.1016/j.jmb.2019.04.045.

[51] Carsen Stringer et al. "Cellpose: a generalist algorithm for cellular segmentation". In: *Nature Methods* 18.1 (Dec. 2020), pp. 100–106. ISSN: 1548-7105. DOI: 10.1038/s41592-020-01018-x. URL: http://dx.doi.org/10.1038/s41592-020-01018-x.

[52] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome Biology* 19.1 (Feb. 2018). DOI: 10.1186/s13059-017-1382-0. URL: https://doi.org/10.1186/s13059-017-1382-0.

[53] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[54] D.L. Donoho. "De-noising by soft-thresholding". In: *IEEE Transactions on Information Theory* 41.3 (May 1995), pp. 613–627. ISSN: 0018-9448. DOI: 10.1109/18.382009. URL: http://dx.doi.org/10.1109/18.382009.

[55] Janne Korhonen et al. "MOODS: fast search for position weight matrix matches in DNA sequences". In: *Bioinformatics* 25.23 (Sept. 2009), pp. 3181–3182. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp554. URL: http://dx.doi.org/10.1093/bioinformatics/btp554.

[56] David R. Cox. *The Statistical Analysis of Series of Events*. Cambridge: Cambridge University Press, 1967.

Appendix A

Learning representations from mass spectra for peptide property prediction

Authors: Michael Murphy, Kevin Yang, Stefanie Jegelka, Ernest Fraenkel

This work was presented at the 2022 ICML workshop on computational biology. I was the primary contributor: I developed the algorithms, carried out the analysis, and wrote the manuscript. I conceived of the idea when interviewing with Kevin Yang. Kevin Yang, Stefanie Jegelka, and Ernest Fraenkel provided commentary on the manuscript and input throughout.

A.1 Abstract

Peptide molecules have long been viewed as promising candidates for design of novel therapeutics, and are likely to benefit from ML-guided approaches. However, learning to predict biological properties of peptides often suffers from a scarcity of labelled training data. We demonstrate mass spectrometry, which provides high-throughput, multidimensional biophysical measurements of peptides, can be used to learn effective representations for peptide property prediction. Specifically, our pretext task asks to identify masked residues of a peptide sequence using its mass spectrum. This yields an encoder that we can then apply to any peptide sequence, irrespective of whether we have spectra for it. Our approach is competitive with a state-of-the-art evolutionary pretext task on a number of downstream tasks, and requires orders of magnitude fewer pretraining examples.

A.2 Introduction

The functional diversity and ease of synthesis of peptide molecules makes them attractive candidates for drug development [1]. However, an exponentially-large search space and often-difficult experimental protocols make it challenging to design peptides with specific properties via traditional laboratory techniques. Machine learning methods promise to fill this gap [2, 3, 4]: given a surrogate model mapping sequence to biological function, we

can screen large numbers of peptides in-silico and prioritize candidates for experimental followup. Unfortunately, many property prediction tasks of therapeutic relevance suffer from a scarcity of labelled sequences [2]. This calls for schemes such as multi-task learning [5], active learning-guided experimental design [6], or self-supervised pretraining from unlabelled protein sequences [7, 8].

The state-of-the-art in pretraining for amino acid sequences applies the *masked language modelling* task introduced by Devlin et al. [9] to large protein databases: a subset of the residues in a protein sequence are masked, and the model is asked to predict these from the rest. The pretraining signal here is *evolutionary*: protein sequences are not random, but rather arise through natural selection to maximize fitness. The masking pretext can therefore be interpreted as predicting which residue(s) would maximize biological activity of the protein encoded by the sequence.

While evolutionary pretraining enjoys a biologically plausible objective for protein sequences, typically hundreds of amino acids long, it may not be optimal for *peptides*, which are at most tens of amino acids long, and are not usually evolutionarily selected for function in isolation from a larger protein. This precludes training a similar masking pretext on peptide-length sequences without additional context, as they lack sufficient information to identify the masked residue. Learning to extract this additional context from proteins incurs a high computational cost, requiring tens to hundreds of millions of long sequences [8, 7]. We resolve these issues by incorporating an information-rich, plentiful – yet to our knowledge, previously-unexploited – data modality for pretraining of peptide property prediction tasks: *mass spectrometry*.

A typical proteomics experiment produces mass spectra of tens of thousands of peptides [10]. Following algorithmic annotation, each spectrum represents a histogram of ions formed by fragmentation of an ionized peptide, which describes the propensity of the peptide to cleave at each bond along the backbone. This depends on a number of related physical properties, including the identities of the amino acids, the distribution of charge along the sidechains and backbone [11], and the peptide's secondary structure [12]. All these properties are also central to biological activities – and, we conjecture, are more directly captured by the *biophysical* signals measured in mass spectrometry.

In short, this work makes the following contributions:

- We identify mass spectrometry as a modality capturing information relevant to peptide property prediction;
- We develop a pretext task that uses mass spectra to guide learning of a representation that can be used for peptide sequences directly, and is hence applicable to peptides that lack spectra; and
- We show that representations learned with this pretext task are competitive with evolutionary pretraining on a number of peptide property prediction tasks, using far fewer pretraining examples.

A.3 Related work

There is substantial recent interest in evolutionary learning of protein representations using large language models, primarily transformers [7, 13, 14], but also convolutional neural networks [8]. These have recently been applied to peptide property prediction [15]. Others pretrain for protein tasks by predicting other modalities from sequence, including 3D structure [16, 17] and functional annotations [18]; but to our knowledge, mass spectrometry has not yet been used for this purpose. We also draw inspiration from successful application of deep learning for NLP to other tasks in mass spectrometry, including predicting spectra from sequence [19, 20] and sequence from spectra [21, 22].

A.4 Methods

A.4.1 Pretext task

Our objective is similar to the masked language model in [8] and [7], in which we randomly mask a single residue from the peptide sequence and require our model to correctly impute it.[1] However, short peptide sequences alone do not provide sufficient context for this task. We therefore additionally condition on the entire observed mass spectrum in addition to the remainder of the sequence; which, as we later show, *does* contain sufficient information to identify the masked amino acid. By returning an intermediate representation of the peptide sequence prior to when it is merged with the spectrum, we can then apply the resulting model to *any* peptide sequence – not just those for which we have spectra.

A.4.2 Model architecture

Our architecture, shown in Figure A.1, comprises two paired encoders: one for the sequence modality, and one for the spectrum modality. The sequence encoder follows the same architecture as the CARP model described in [8]; briefly, it converts each amino acid symbol to an 8-dimensional embedding vector, and then passes this sequence of embeddings through a ByteNet dilated CNN as developed in [23], which yields a *sequence-length* encoding at its output.

We represent a mass spectrum of a length-L peptide as an $(L-1) \times K$-dimensional table of probabilities of each of K ion types arising from cleavage of $L-1$ bonds, which we compute by normalizing the observed counts across annotated peaks to sum to 1. We also concatenate at each bond two scalar-valued properties of the spectrum as a whole: its observed electric charge prior to fragmentation, and the collision energy at which it was measured. This is passed into another ByteNet encoder, yielding at its output one embedding vector per bond. The outputs of the two encoders are concatenated feature-wise (padding the bond encodings to length L) and passed into a 2-layer ReLU classifier applied independently at each position,

[1] We initially tried a simpler pretext of predicting spectra from sequence: using only a sequence encoder, and yielding a logit per ion type instead of per amino acid in Figure A.1. While this performed well on the pretext (mean test $R^2 = 0.93$), we found it less effective on downstream tasks than our masking objective.

yielding a vector of logits per each of t amino acids. We then index into the prediction at the masked token and minimize cross-entropy loss against the true residue.

For both encoders we use the same depth and width as the smallest pretrained model in [8], `CARP-600k`, which uses $n = 16$ layers of $d = 128$-dimensional ByteNet blocks. We train using minibatches of 512 peptide-spectrum pairs with Adam [24] (learning rate $= 5 \times 10^{-4}$), early-stopping on validation cross-entropy after 126 epochs.

A.4.3 Pretext dataset

We use a library of mass spectra generated in Part I of the ProteomeTools project [25]. This comprises 2,980,009 annotated mass spectra of 391,273 unique peptides derived from the human proteome, composed of the 20 canonical amino acids plus two post-translational modifications (methionine sulfoxide and carbamidomethyl-cysteine). The redundancy per peptide arises through different combinations of charge states, collision energies, and modifications, each of which generally yield distinct spectra. These spectra are provided as lists of (mass, intensity) tuples, each of which is annotated with the bond and one of K possible types of the respective fragment. We use an 85/5/10 train/validation/test split; to avoid leakage, we cluster sequences using CD-HIT [26] (threshold 0.5, word length 3) and randomly assign entire clusters to each split.

A.4.4 Peptide property prediction datasets

We identified a number of datasets of peptide sequences, labelled either positive or negative for some biological property, as representative of potential objectives for peptide design. Where applicable, we only include sequences comprising the 20 canonical amino acids, with $5 \leq L \leq 100$:

- Mitochondrial Targeting. Zarin et al. [27] provide annotations of 5,348 N-terminal intrinsically-disordered regions (IDRs) identified in a screen for mitochondrial targeting in yeast. From these sequences we select 160 positive and 3,960 negative examples.
- Cdc28 Binding. Zarin et al. [27] also indicate whether the same IDRs are substrates of the kinase Cdc28: this gives 80 positive examples and 4040 negative examples.
- Signal Peptide. SignalP [28] is a high-quality repository of annotated signal peptide sequences derived from eukaryotes and prokaryotes. For simplicity we consider only the SignalP 6.0 training set and ignore type annotations for the prokaryotic sequences, resulting in a binary classification problem of 15,625 positives against 4,665 negatives.
- MHC Binding. The Dana-Farber Repository for Machine Learning in Immunology [29] provides peptide sequences that bind a number of human and mouse MHC-II complexes. We construct as our positive class the unique 'binding' peptide sequences across the union of training sets listed at http://projects.met-hilab.org/DFRMLI/HTML/natural.php, and the union of 'non-binding' peptides for the negative class; discarding sequences appearing in both to yield 9,720 positive examples and 6,945 negative examples.

Within each task we carry out 3-fold nested cross-validation, again splitting via CD-HIT clustering.

A.4.5 Downstream tasks

To apply our model to the downstream tasks, we discard the spectrum encoder and final classifier, evaluate the sequence encoder on the input peptide sequence, and learn a new classifier on the resulting embeddings. Because the encoder yields a sequence-length representation, our classifier pools across positions via an attention layer [30], then applies a 2-layer ReLU binary classifier.

We compare to four baselines. (1) `CARP-600k` [8] is employed as representative of the state-of-the-art in evolutionary pretraining. This model has essentially the same architecture as our sequence encoder – permitting direct comparison of pretext tasks without confounding from architecture – but is trained on far more data: 41.5 million full-length protein sequences from UniRef50 [31]. (2) We also include our model initialized from random without pretraining. Finally, we use two simple models: (3) a length-averaged prediction of a linear classifier applied individually to each amino acid; and (4) a 3-layer, 128-wide CNN with ReLU activations, kernel width of 5, and length-wise average-pooling, prior to a linear output layer.

For pretrained models, we test both freezing the encoder weights and training only the final classifier, and fine-tuning the entire network. All models use Adam (batch size 256, learning rate 5×10^{-4}), early-stopping on validation AUC.

A.5 Results and Discussion

A.5.1 Pretext accuracy

Our method identifies randomly-masked amino acids with 69.9% accuracy on the test set. In comparison, evaluating the pretrained evolutionary model on peptide sequences achieves only 10.4% accuracy: this is unsurprising, as the global protein context on which it depends is missing. To confirm the spectral information is actually used, we also tried solving our pretext using the sequence alone: this peaked at a maximal validation accuracy of 18% and then proceeded to overfit, indicating masking alone is insufficient for peptides and spectra are indeed necessary.

Figure A.2 shows a per-amino-acid confusion matrix. Reassuringly, errors tend to be structurally-similar amino acids: in particular the branched-chain amino acids (I = isoleucine, L = leucine, V = valine) and two of the three aromatic amino acids (F = phenylalanine, Y = tyrosine). We also see methionine (M) and its modified form (m) are *not* frequently confused. This suggests a potential blind spot of evolutionary pretraining, which does not separately represent modified and unmodified amino acids.

A.5.2 Interpretation of embeddings

We embed each sequence in our test set, average-pool lengthwise, and visualize the result via UMAP [32] in Figure A.3. This indicates our sequence representation captures known determinants of peptide fragmentation. Length dependence is apparent, as is clustering according to: the number of basic amino acids, which determines the maximum charge a peptide can carry and where it localizes [11]; the presence of proline (P), which bends the peptide [33]; and the identity of the C-terminal amino acid, which reflects a selection bias from using trypsin to digest proteins into peptides.

A.5.3 Downstream performance

Test AUC of our model and the baselines for the four tasks considered are shown in Table A.1. On the Cdc28 Binding and Signal Peptide tasks, a model pretrained on mass spectrometry is competitive with the state-of-the-art evolutionary approach. Our approach also outperforms state-of-the-art on Mitochondrial Targeting. We suggest this is due to our choice of data: mass spectral peak intensities are substantially determined by the distribution of electric charge within the peptide, which is also known to determine mitochondrial targeting [27]. However, our approach fares poorly on MHC Binding; the lower performance overall on that task may have resulted from our decision to pool sequences across different MHC complexes.

The evolutionary pretext, trained on proteins, proves effective for peptides. This might be due to the ByteNet architecture, whose first few layers of convolutional filters necessarily detect peptide-sized features. But this evolutionary pretext also benefits from a much larger corpus than our peptide task: `CARP-600k` is trained on about 100× more sequences than our model, each much longer than a peptide.

Model		Mito	Cdc28	SignalP	MHC
CARP-600k	FR	0.87	0.73	**0.99**	0.72
	FT	0.86	**0.78**	**0.99**	0.73
Linear		0.86	0.56	0.82	0.67
3-layer CNN		0.85	0.71	0.92	**0.75**
Random init.	FR	0.85	0.71	**0.99**	0.72
	FT	0.83	0.74	0.94	0.68
MS pretrained	FR	**0.89**	**0.78**	0.97	0.70
	FT	**0.89**	0.71	**0.99**	0.72

Table A.1: Averaged 3-fold test AUC on downstream tasks, for: CARP-600k (SOTA); linear and CNN baselines; our model, initalized from random; and pretraining on the MS pretext. 'FR' only trains the final classifier; 'FT' additionally trains the encoder.

A.6 Conclusion

Here we show existing published mass spectrometry data can be used to derive representations for peptide property prediction that are competitive with a state-of-the-art evolutionary pretext, while using far fewer sequences. Evolutionary and mass-spectral pretraining need not be mutually exclusive: both enjoy plentiful data and may offer complementary views of peptide and protein structure – particularly for modified amino acids, which are not explicitly represented in evolutionary data, yet strongly influence protein structure and function [34] and are represented with greater diversity in other ProteomeTools releases [35]. Integration of these two modalities is a promising avenue for further exploration, especially for peptide design tasks in which modifications or peptidomimetics [36] are included in the sequence search space.

References

[1] Keld Fosgerau and Torsten Hoffmann. "Peptide therapeutics: current status and future directions". In: *Drug Discovery Today* 20.1 (Jan. 2015), pp. 122–128. DOI: 10.1016/j.drudis.2014.10.003. URL: https://doi.org/10.1016/j.drudis.2014.10.003.

[2] Xumin Chen et al. "Sequence-based peptide identification, generation, and property prediction with deep learning: a review". In: *Molecular Systems Design and Engineering* 6.6 (2021), pp. 406–428. DOI: 10.1039/d0me00161a. URL: https://doi.org/10.1039/d0me00161a.

[3] Fangping Wan, Daphne Kontogiorgos-Heintz, and Cesar de la Fuente-Nunez. "Deep generative models for peptide design". In: *Digital Discovery* (2022). DOI: 10.1039/d1dd00024a. URL: https://doi.org/10.1039/d1dd00024a.

[4] Carly K Schissel et al. "Interpretable Deep Learning for De Novo Design of Cell-Penetrating Abiotic Polymers". In: *bioRxiv* (2020). DOI: 10.1101/2020.04.10.036566. URL: https://www.biorxiv.org/content/10.1101/2020.04.10.036566v1.

[5] Yanjun Qi et al. "A Unified Multitask Architecture for Predicting Local Protein Properties". In: *PLoS ONE* 7.3 (Mar. 2012). Ed. by Christos A. Ouzounis, e32235. DOI: 10.1371/journal.pone.0032235. URL: https://doi.org/10.1371/journal.pone.0032235.

[6] Kevin K. Yang, Zachary Wu, and Frances H. Arnold. "Machine-learning-guided directed evolution for protein engineering". In: *Nature Methods* 16.8 (July 2019), pp. 687–694. DOI: 10.1038/s41592-019-0496-6. URL: https://doi.org/10.1038/s41592-019-0496-6.

[7] Alexander Rives et al. "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences". In: *Proceedings of the National Academy of Sciences* 118.15 (Apr. 2021). DOI: 10.1073/pnas.2016239118. URL: https://doi.org/10.1073/pnas.2016239118.

[8] Kevin K Yang, Alex Xijie Lu, and Nicolo Fusi. "Convolutions are competitive with transformers for protein sequence pretraining". In: *ICLR2022 Machine Learning for Drug Discovery*. 2022. URL: https://openreview.net/forum?id=3i7WPak2sCx.

[9] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018. DOI: 10.48550/ARXIV.1810.04805. URL: https://arxiv.org/abs/1810.04805.

[10] Hanno Steen and Matthias Mann. "The abc's (and xyz's) of peptide sequencing". In: *Nature Reviews Molecular Cell Biology* 5.9 (Sept. 2004), pp. 699–711. DOI: 10.1038/nrm1468. URL: https://doi.org/10.1038/nrm1468.

[11] Bela Paizs and Sandor Suhai. "Fragmentation pathways of protonated peptides". In: *Mass Spectrometry Reviews* 24.4 (2005), pp. 508–548. DOI: 10.1002/mas.20024. URL: https://doi.org/10.1002/mas.20024.

[12] George Tsaprailis et al. "Influence of Secondary Structure on the Fragmentation of Protonated Peptides". In: *Journal of the American Chemical Society* 121.22 (May 1999), pp. 5142–5154. DOI: 10.1021/ja982980h. URL: https://doi.org/10.1021/ja982980h.

[13] Ahmed Elnaggar et al. "ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), pp. 1–1. DOI: 10.1109/tpami.2021.3095381. URL: https://doi.org/10.1109/tpami.2021.3095381.

[14] Roshan Rao et al. "Evaluating Protein Transfer Learning with TAPE". In: *Advances in Neural Information Processing Systems*. 2019.

[15] William Dee. "LMPred: predicting antimicrobial peptides using pre-trained language models and deep learning". In: *Bioinformatics Advances* 2.1 (Jan. 2022). Ed. by Michael Gromiha. DOI: 10.1093/bioadv/vbac021. URL: https://doi.org/10.1093/bioadv/vbac021.

[16] Tristan Bepler and Bonnie Berger. "Learning protein sequence embeddings using information from structure". In: (2019). DOI: 10.48550/ARXIV.1902.08661. URL: https://arxiv.org/abs/1902.08661.

[17] Ningyu Zhang et al. "OntoProtein: Protein Pretraining With Gene Ontology Embedding". In: *arXiv preprint arXiv:2201.11147* (2022).

[18] Zuobai Zhang et al. *Protein Representation Learning by Geometric Structure Pretraining*. 2022. DOI: 10.48550/ARXIV.2203.06125. URL: https://arxiv.org/abs/2203.06125.

[19] Xie-Xuan Zhou et al. "pDeep: Predicting MS/MS Spectra of Peptides with Deep Learning". In: *Analytical Chemistry* 89.23 (Nov. 2017), pp. 12690–12697. DOI: 10.1021/acs.analchem.7b02566. URL: https://doi.org/10.1021/acs.analchem.7b02566.

[20] Siegfried Gessulat et al. "Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning". In: *Nature Methods* 16.6 (May 2019), pp. 509–518. DOI: 10.1038/s41592-019-0426-7. URL: https://doi.org/10.1038/s41592-019-0426-7.

[21] Melih Yilmaz et al. "De novo mass spectrometry peptide sequencing with a transformer model". In: (Feb. 2022). DOI: 10.1101/2022.02.07.479481. URL: https://doi.org/10.1101/2022.02.07.479481.

[22] Rui Qiao et al. "Computationally instrument-resolution-independent de novo peptide sequencing for high-resolution devices". In: *Nature Machine Intelligence* 3.5 (Mar. 2021), pp. 420–425. DOI: 10.1038/s42256-021-00304-3. URL: https://doi.org/10.1038/s42256-021-00304-3.

[23] Nal Kalchbrenner et al. "Neural Machine Translation in Linear Time". In: *CoRR* abs/1610.10099 (2016). arXiv: 1610.10099. URL: http://arxiv.org/abs/1610.10099.

[24] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2014. DOI: 10.48550/ARXIV.1412.6980. URL: https://arxiv.org/abs/1412.6980.

[25] Daniel P Zolg et al. "Building ProteomeTools based on a complete synthetic human proteome". In: *Nature Methods* 14.3 (Jan. 2017), pp. 259–262. DOI: 10.1038/nmeth.4153. URL: https://doi.org/10.1038/nmeth.4153.

[26] Limin Fu et al. "CD-HIT: accelerated for clustering the next-generation sequencing data". In: *Bioinformatics* 28.23 (Oct. 2012), pp. 3150–3152. DOI: 10.1093/bioinformatics/bts565. URL: https://doi.org/10.1093/bioinformatics/bts565.

[27] Taraneh Zarin et al. "Identifying molecular features that are associated with biological function of intrinsically disordered protein regions". In: *eLife* 10 (Feb. 2021). DOI: 10.7554/elife.60220. URL: https://doi.org/10.7554/elife.60220.

[28] Felix Teufel et al. "SignalP 6.0 predicts all five types of signal peptides using protein language models". In: *Nature Biotechnology* (Jan. 2022). DOI: 10.1038/s41587-021-01156-3. URL: https://doi.org/10.1038/s41587-021-01156-3.

[29] Guang Lan Zhang et al. "Dana-Farber repository for machine learning in immunology". In: *Journal of Immunological Methods* 374.1-2 (Nov. 2011), pp. 18–25. DOI: 10.1016/j.jim.2011.07.007. URL: https://doi.org/10.1016/j.jim.2011.07.007.

[30] Ashish Vaswani et al. "Attention Is All You Need". In: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.

[31] B. E. Suzek et al. "UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches". In: *Bioinformatics* 31.6 (Nov. 2014), pp. 926–932. DOI: 10.1093/bioinformatics/btu739. URL: https://doi.org/10.1093/bioinformatics/btu739.

[32] Leland McInnes et al. "UMAP: Uniform Manifold Approximation and Projection". In: *Journal of Open Source Software* 3.29 (Sept. 2018), p. 861. DOI: 10.21105/joss.00861. URL: https://doi.org/10.21105/joss.00861.

[33] T. Vaisar and J. Urban. "Probing Proline Effect in CID of Protonated Peptides". In: *Journal of Mass Spectrometry* 31.10 (Oct. 1996), pp. 1185–1187. DOI: 10.1002/(sici)1096-9888(199610)31:10<1185::aid-jms396>3.0.co;2-q. URL: https://doi.org/10.1002/(sici)1096-9888(199610)31:10%3C1185::aid-jms396%3E3.0.co;2-q.

[34] Matthias Mann and Ole N. Jensen. "Proteomic analysis of post-translational modifications". In: *Nature Biotechnology* 21.3 (Mar. 2003), pp. 255–261. DOI: 10.1038/nbt0303-255. URL: https://doi.org/10.1038/nbt0303-255.

[35] Daniel Paul Zolg et al. "ProteomeTools: Systematic Characterization of 21 Post-translational Protein Modifications by Liquid Chromatography Tandem Mass Spectrometry (LC-MS/MS) Using Synthetic Peptides". In: *Molecular and Cellular Proteomics* 17.9 (Sept. 2018), pp. 1850–1863. DOI: 10.1074/mcp.tir118.000783. URL: https://doi.org/10.1074/mcp.tir118.000783.

[36] Annarita Del Gatto et al. "Editorial: Peptidomimetics: Synthetic Tools for Drug Discovery and Development". In: *Frontiers in Chemistry* 9 (Nov. 2021). DOI: 10.3389/fchem.2021.802120. URL: https://doi.org/10.3389/fchem.2021.802120.

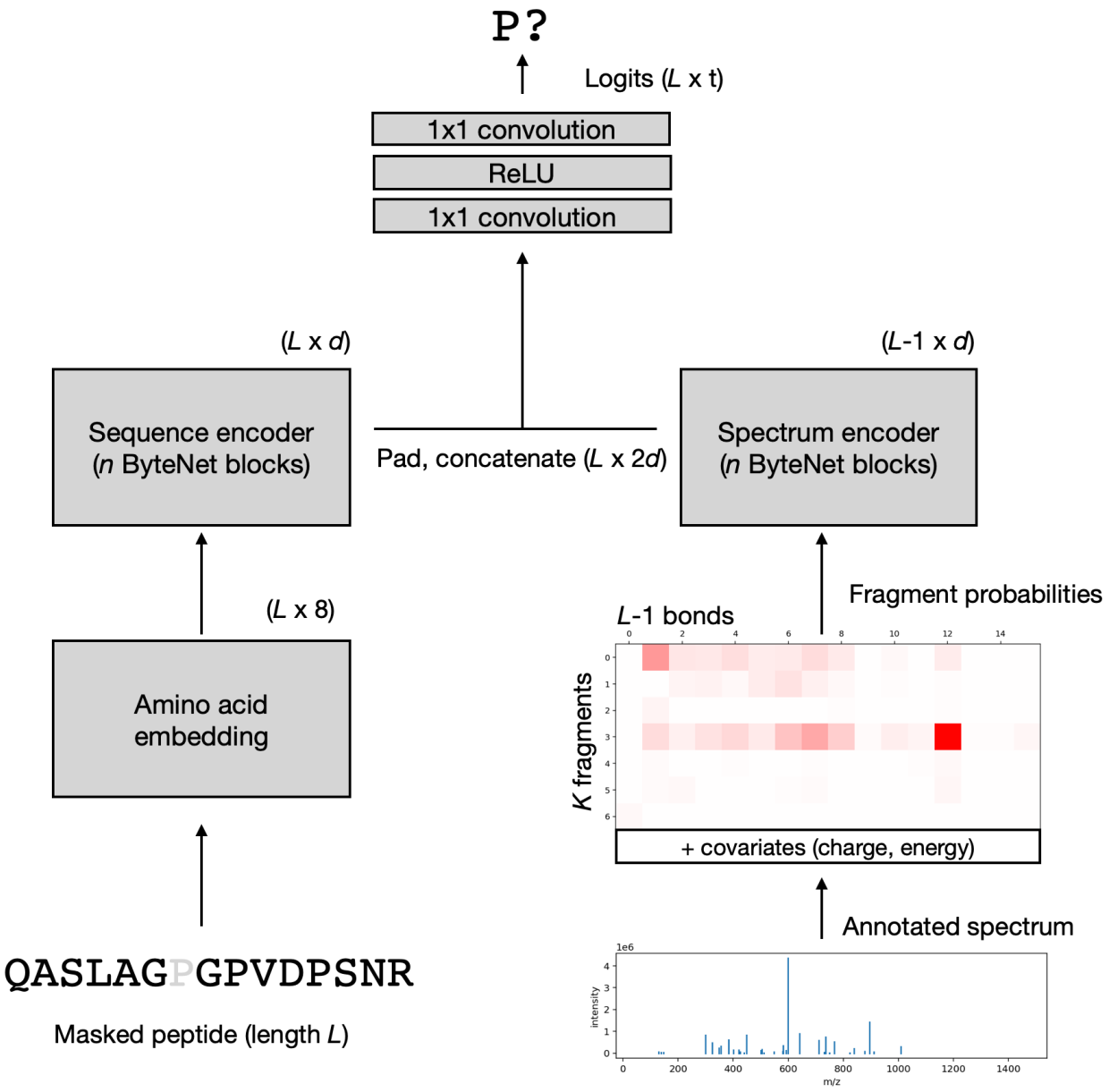

Figure A.1: Our model's architecture. We return the $L \times d$ output of the sequence encoder as our embeddings. See Yang, Lu, and Fusi [8] for a more in-depth exposition of the ByteNet architecture.

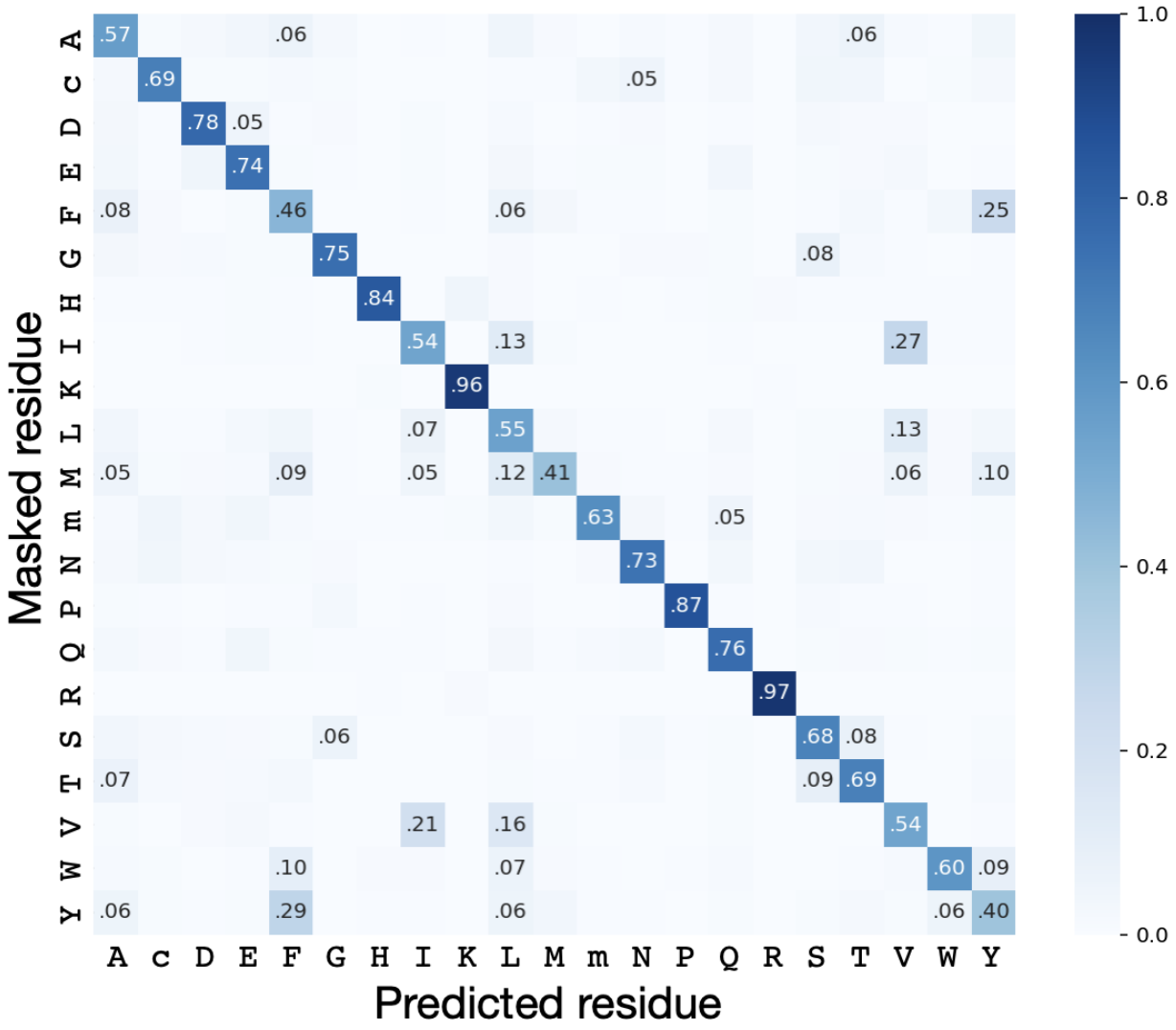

Figure A.2: Confusion matrix for the pretext task between masked (rows) and predicted (columns) amino acids. Color indicates $P(\text{column} \mid \text{row})$. Entries > 0.05 are labelled.

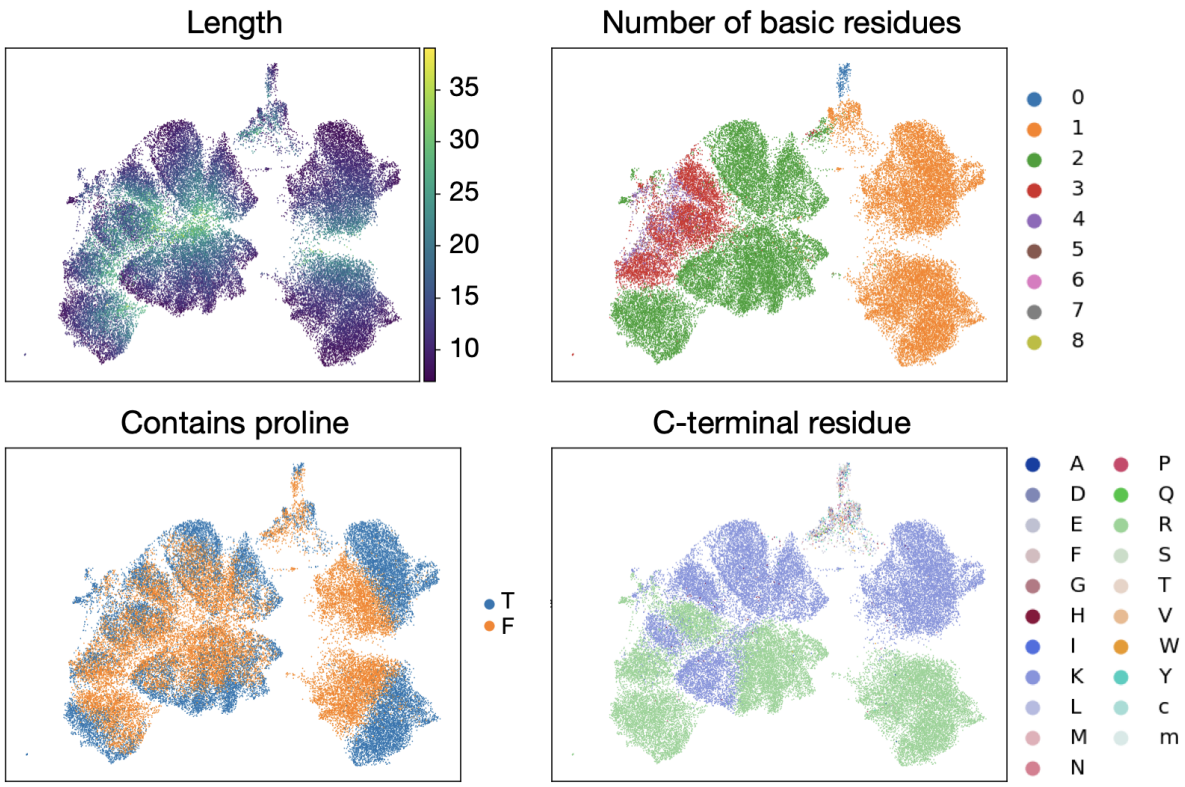

Figure A.3: UMAP visualization of peptide embeddings, colored according to properties known to influence fragmentation.

Appendix B

Statistical analysis of single-cell metabolomics

I contributed this section to the review paper 'Single cell metabolism: current and future trends', published in Metabolomics 2022.

While the statistical analysis of single-cell metabolomics remains in its infancy, single-cell transcriptomics have matured to the point where sophisticated statistical tools for high-dimensional data can be routinely applied as part of standard data processing workflows [1]. This is largely facilitated by the widespread adoption among the scRNA community of mature open-source data analysis packages written in modern scripting languages, such as ScanPy [2] for Python, its extension SquidPy for spatial omics [3], and Seurat [4] for R. Such tools can also be applied to MS-based single-cell metabolomics data once mass spectra are converted to intensities of annotated metabolites. Rappez et al [5] used ScanPy to perform statistical analyses on a dataset of 740 metabolites from 29,738 hepatocytes, including cell-to-cell normalization, nonlinear batch correction, UMAP dimensionality reduction [6], Leiden clustering, and pseudotime trajectory analysis [7]. This revealed three clusters of cells corresponding to homeostasis, steatosis, and an intermediate metabolic state, for which marker metabolites were identified via hypothesis tests of differential expression. The ScanPy package was also used in [8] to carry out clustering and differential expression testing for mass spectra of single organelles, and in [9] for differential expression analysis, dimensionality reduction, and clustering of mass spectra of single hepatocyte nuclei in-situ.

However, single-cell metabolomics data differs from other modalities in ways that may present obstacles to further cross-application of existing statistical methods. Firstly, MS, optical, and vibrational metabolomics data are natively continuous-valued signals, whereas sequencing data is count-valued and often sparse. The meanings of zeroes also differ between transcriptomics and MS-based metabolomics: while the technical versus biological origins of sparsity in scRNA have been the subject of debate [10, 11], metabolite peaks can fail to be detected in mass spectra for computational reasons: e.g. if they insufficiently exceed background spectral noise during peak detection [12], or if they are mis-annotated due to mass drift [13]. While such events may be digitally recorded as zeroes, they are better understood and modelled as a form of random missingness [14]. Methods based upon MS imaging of single cells also risk co-sampling of neighboring cells: although the authors of

[5] indicate this minimally affected their analysis of cultured hepatocytes, this may bias estimates of spatial autocorrelation of metabolites in tissues where cells are tightly packed relative to the MS spatial resolution. Finally, a significant promise of single-cell omics is its potential (through sufficiently large sample sizes) to infer regulatory relationships de-novo between biomolecules. Many algorithms toward this end have been developed for single-cell RNA data [15, 16, 17], as well as for mass cytometry [18, 19]. While in principle such methods could also be applied toward learning metabolic networks from data, we caution these may not be robust to sources of (co)variation specific to MS-based single-cell metabolomics, such as in-source fragmentation [20], region-dependent ion suppression [21], or global metabolic shifts arising from cell manipulation [22]: these may induce correlations between metabolites that are primarily technical as opposed to biological in origin.

References

[1] Malte D Luecken and Fabian J Theis. "Current best practices in single-cell RNA-seq analysis: a tutorial". In: *Molecular Systems Biology* 15.6 (June 2019). DOI: 10.15252/msb.20188746. URL: https://doi.org/10.15252/msb.20188746.

[2] F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome Biology* 19.1 (Feb. 2018). DOI: 10.1186/s13059-017-1382-0. URL: https://doi.org/10.1186/s13059-017-1382-0.

[3] Giovanni Palla et al. "Squidpy: a scalable framework for spatial omics analysis". In: *Nature Methods* 19.2 (Jan. 2022), pp. 171–178. DOI: 10.1038/s41592-021-01358-2. URL: https://doi.org/10.1038/s41592-021-01358-2.

[4] Yuhan Hao et al. "Integrated analysis of multimodal single-cell data". In: *Cell* 184.13 (June 2021), 3573–3587.e29. DOI: 10.1016/j.cell.2021.04.048. URL: https://doi.org/10.1016/j.cell.2021.04.048.

[5] Luca Rappez et al. "SpaceM reveals metabolic states of single cells". In: *Nature Methods* 18.7 (July 2021), pp. 799–805. DOI: 10.1038/s41592-021-01198-0. URL: https://doi.org/10.1038/s41592-021-01198-0.

[6] Etienne Becht et al. "Dimensionality reduction for visualizing single-cell data using UMAP". In: *Nature Biotechnology* 37.1 (Dec. 2018), pp. 38–44. DOI: 10.1038/nbt.4314. URL: https://doi.org/10.1038/nbt.4314.

[7] Cole Trapnell et al. "The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells". In: *Nature Biotechnology* 32.4 (Mar. 2014), pp. 381–386. DOI: 10.1038/nbt.2859. URL: https://doi.org/10.1038/nbt.2859.

[8] Daniel C. Castro et al. "Image-guided MALDI mass spectrometry for high-throughput single-organelle characterization". In: *Nature Methods* 18.10 (Sept. 2021), pp. 1233–1238. DOI: 10.1038/s41592-021-01277-2. URL: https://doi.org/10.1038/s41592-021-01277-2.

[9] Zhiyuan Yuan et al. "SEAM is a spatial single nuclear metabolomics method for dissecting tissue microenvironment". In: *Nature Methods* 18.10 (Oct. 2021), pp. 1223–1232. DOI: 10.1038/s41592-021-01276-3. URL: https://doi.org/10.1038/s41592-021-01276-3.

[10] Valentine Svensson. "Droplet scRNA-seq is not zero-inflated". In: *Nature Biotechnology* 38.2 (Jan. 2020), pp. 147–150. DOI: 10.1038/s41587-019-0379-5. URL: https://doi.org/10.1038/s41587-019-0379-5.

[11] Ruochen Jiang et al. "Statistics or biology: the zero-inflation controversy about scRNA-seq data". In: *Genome Biology* 23.1 (Jan. 2022). DOI: 10.1186/s13059-022-02601-5. URL: https://doi.org/10.1186/s13059-022-02601-5.

[12] Theodore Alexandrov. "MALDI imaging mass spectrometry: statistical data analysis and current computational challenges". In: *BMC Bioinformatics* 13.S16 (Nov. 2012). DOI: 10.1186/1471-2105-13-s16-s11. URL: https://doi.org/10.1186/1471-2105-13-s16-s11.

[13] Raphaël La Rocca et al. "Adaptive Pixel Mass Recalibration for Mass Spectrometry Imaging Based on Locally Endogenous Biological Signals". In: *Analytical Chemistry* 93.8 (Feb. 2021), pp. 4066–4074. DOI: 10.1021/acs.analchem.0c05071. URL: https://doi.org/10.1021/acs.analchem.0c05071.

[14] Kieu Trinh Do et al. "Characterization of missing values in untargeted MS-based metabolomics data and evaluation of missing data handling strategies". In: *Metabolomics* 14.10 (Sept. 2018). DOI: 10.1007/s11306-018-1420-2. URL: https://doi.org/10.1007/s11306-018-1420-2.

[15] Sara Aibar et al. "SCENIC: single-cell regulatory network inference and clustering". In: *Nature Methods* 14.11 (Oct. 2017), pp. 1083–1086. DOI: 10.1038/nmeth.4463. URL: https://doi.org/10.1038/nmeth.4463.

[16] Aditya Pratapa et al. "Benchmarking algorithms for gene regulatory network inference from single-cell transcriptomic data". In: *Nature Methods* 17.2 (Jan. 2020), pp. 147–154. DOI: 10.1038/s41592-019-0690-6. URL: https://doi.org/10.1038/s41592-019-0690-6.

[17] Thalia E. Chan, Michael P.H. Stumpf, and Ann C. Babtie. "Gene Regulatory Network Inference from Single-Cell Data Using Multivariate Information Measures". In: *Cell Systems* 5.3 (Sept. 2017), 251–267.e3. DOI: 10.1016/j.cels.2017.08.014. URL: https://doi.org/10.1016/j.cels.2017.08.014.

[18] Karen Sachs et al. "Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data". In: *Science* 308.5721 (Apr. 2005), pp. 523–529. DOI: 10.1126/science.1105809. URL: https://doi.org/10.1126/science.1105809.

[19] Shu Wang et al. "Inferring reaction network structure from single-cell, multiplex data, using toric systems theory". In: *PLOS Computational Biology* 15.12 (Dec. 2019). Ed. by Pedro Mendes, e1007311. DOI: 10.1371/journal.pcbi.1007311. URL: https://doi.org/10.1371/journal.pcbi.1007311.

[20] Marjan Dolatmoradi, Laith Z. Samarah, and Akos Vertes. "Single-Cell Metabolomics by Mass Spectrometry: Opportunities and Challenges". In: *Analysis and Sensing* 2.1 (Nov. 2021). DOI: 10.1002/anse.202100032. URL: https://doi.org/10.1002/anse.202100032.

[21] Adam J. Taylor, Alex Dexter, and Josephine Bunch. "Exploring Ion Suppression in Mass Spectrometry Imaging of a Heterogeneous Tissue". In: *Analytical Chemistry* 90.9 (Feb. 2018), pp. 5637–5645. DOI: 10.1021/acs.analchem.7b05005. URL: https://doi.org/10.1021/acs.analchem.7b05005.

[22] Elizabeth M. Llufrio et al. "Sorting cells alters their redox state and cellular metabolome". In: *Redox Biology* 16 (June 2018), pp. 381–387. DOI: 10.1016/j.redox.2018.03.004. URL: https://doi.org/10.1016/j.redox.2018.03.004.

Appendix C

Differentiable min-cuts for predicting small molecule fragmentation

This section constitutes unpublished followup work to Chapter 3 of this thesis.

The lack of training data available for the small molecule spectrum prediction problem suggests we should incorporate stronger inductive biases. Here I pressent an approach for 'neuralizing' bond-breaking. Specifically, I propose we model the (unnormalized) likelihood of a fragmentation transition $\mathcal{G} \to \mathcal{H}$ as decomposing over the bonds that are broken:

$$\begin{aligned} P(G \to H) &\propto \prod_{(i,j)\in G, i\in H, j\notin H} P(\text{bond } i-j \text{ breaks} \mid G) \\ &\doteq \prod_{(i,j)\in G} p_{ij}^{x_i(1-x_j)} \\ -\log P(G \to H) &= \sum_{(i,j)\in G} w_{ij} x_i (1-x_j) \end{aligned}$$

where $w_{ij} \doteq -\log p_{ij}$ is large when bond (i,j) is unlikely to break, and $x_i \doteq 1_{i\in H}$. We see this negative-log-likelihood matches the cost of a cut induced by an indicator vector $x \in \{0,1\}^G$. If we had a hypothetical dataset of ground-truth substructures H generated by fragmentation of different molecules G, we could parametrize $W = W(G;\theta)$ and estimate θ by minimizing a cross entropy:

$$\theta^* = \arg\min_\theta \sum_G \sum_{H\subseteq G} p_H^G \sum_{(i,j)\in G} w_{ij}(G;\theta) x_i^H (1-x_j^H) + \log \mathcal{Z}(G;\theta)$$

where p_H^G represent the probability of substructure H arising from fragmentation of G, and $\mathcal{Z}(G;\theta)$ is a normalization constant.

However, in mass spectrometry we only observe chemical formulas $b^H = A^G x^H$, where A^G is an elements-by-nodes indicator matrix (and associated probabilities p_b^G in the form of peak heights); the substructures are latent, and generally combinatorially many substructures will be compatible with a given formula. We might therefore marginalize out all such

substructures:

$$\theta^* = \arg\min_\theta \sum_G \sum_{b \leq A^G 1} p_b^G \log \sum_{x: A^G x = b} \exp \sum_{(i,j) \in G} w_{ij}(G;\theta) x_i (1 - x_j) + \log \mathcal{Z}(G;\theta)$$

This is expensive. To efficiently learn $W(G;\theta)$, I propose we only consider the *most likely* substructure assignment, which is the cost of a min-cut compatible with b (recalling this is a minimization of a negative log likelihood):

$$\theta^* \approx \arg\min_\theta \sum_G \sum_{b \leq A^G 1} p_b^G \min_{x \in \{0,1\}^G : A^G x = b} \sum_{(i,j) \in G} w_{ij}(G;\theta) x_i (1 - x_j) + \log \mathcal{Z}(G;\theta)$$

We may relax the inner min-cut problem as a convex quadratic program with an affine constraint on the infinity ball. Rescaling $x \to 2x - 1$, $b \to 2b - A1$:

$$\begin{aligned} \min_x & \frac{1}{2} x^T L x + c^T x \\ s.t.\ & Ax = b \\ & \|x\|_\infty \leq 1 \end{aligned}$$

where $L = \text{diag}(W1) - W$ is the graph Laplacian matrix, and $c \in \mathbb{R}^G$ is a small $\mathcal{N}(0, \epsilon)$ vector added to break symmetries. This is differentiable convex optimization; the forward pass can be implemented using `cvxpylayers`, and the backward pass can be computed via implicit differentiation of the KKT conditions. Unfortunately, `cvxpylayers` uses a CPU-bound solver, so training is very slow. I therefore have derived equations for gradient ascent on the dual problem, which turns out to be a LASSO problem that is easily parallelized on GPU.

We first must address a major problem here, which would prevent our subsequent use of the inverse of L: L has a zero eigenvalue. However, that eigenvalue corresponds to the all-ones vector, which cannot lie in the null-space of an indicator matrix – so even though the objective is not strictly convex, the problem should have a unique solution. I make the objective strictly convex by adding an augmented Lagrangian-like term $\frac{1}{2}\|Ax - b\|_2^2$. Rearranging and adding a dummy variable z:

$$\begin{aligned} \min_{x,z} & \frac{1}{2} x^T (L + A^T A) x + (c - A^T b)^T x \\ s.t.\ & Ax = b \\ & \|z\|_\infty \leq 1 \\ & z = x \end{aligned}$$

Now we redefine $L \leftarrow L + A^T A$, $c \leftarrow c - A^T b$, and proceed:

$$\begin{aligned}
\mathcal{L}(x, z, \lambda, \nu) &= \frac{1}{2} x^T L x + c^T x + \nu^T (Ax - b) + \lambda^T (z - x) \\
&= -\nu^T b + \frac{1}{2} x^T L x + (A^T \nu - \lambda + c)^T x + \lambda^T z \\
g(\lambda, \nu) &= \inf_{x,z} \mathcal{L}(x, z, \nu, \lambda) \ s.t. \ \|z\|_\infty \leq 1 \\
&= -\nu^T b + \inf_x \{\frac{1}{2} x^T L x + (A^T \nu - \lambda + c)^T x\} + \inf_z \{\lambda^T z : \|z\|_\infty \leq 1\} \\
&= -\nu^T b + \inf_x \{\frac{1}{2} x^T L x + (A^T \nu - \lambda + c)^T x\} - \sup_z \{(-\lambda)^T z : \|z\|_\infty \leq 1\} \\
&= -\nu^T b - \frac{1}{2} (\lambda - A^T \nu - c)^T L^{-1} (\lambda - A^T \nu - c) - \| - \lambda\|_{\infty^*} \\
&= -\nu^T b - \frac{1}{2} (\lambda - A^T \nu - c)^T L^{-1} (\lambda - A^T \nu - c) - \|\lambda\|_1 \\
-g(\lambda, \nu) &= \frac{1}{2} (\lambda - A^T \nu - c)^T L^{-1} (\lambda - A^T \nu - c) + \nu^T b + \|\lambda\|_1
\end{aligned}$$

Redefining $g \leftarrow -g$ gives us an unconstrained LASSO minimization. We may solve this via (accelerated) proximal gradient descent, using the following update equations:

$$\begin{aligned}
x^t &= L^{-1}(\lambda^t - A^T \nu^t - c) \\
\lambda^{t+1} &= \text{prox}_{\gamma_\lambda \|\cdot\|_1}(\lambda^t - \gamma_\lambda x^t) \\
\nu^{t+1} &= \nu^t - \gamma_\nu (b - Ax^t)
\end{aligned}$$

where $\gamma_\lambda, \gamma_\nu$ are learning rates and $\text{prox}_{\gamma_\lambda \|\cdot\|_1}$ is the soft-thresholding operator.

For the backward pass, we define $\phi(\lambda, \nu; \theta) = 0$ as a fixed-point equation for the proximal gradient iterates:

$$\phi(\lambda, \nu; \theta) = \begin{bmatrix} \lambda - ST_\gamma(\lambda - \gamma x(\lambda, \nu)) \\ b - Ax(\lambda, \nu) \end{bmatrix}$$

where $\phi(\lambda^*, \nu^*; \theta) = 0$ for an optimal dual solution $(\lambda^*, \nu^*) = (\lambda^*, \nu^*)(\theta)$ (using shorthand $\theta \doteq (L, c)$, and ST_γ is the soft-thresholding operator). We now compute gradients $\partial_\theta(\lambda, \nu)$ using implicit differentiation:

$$\begin{aligned}
\partial_\theta \phi(\lambda^*(\theta), \nu^*(\theta); \theta) &= \partial_{\lambda,\nu} \phi(\lambda, \nu; \theta)|_{\lambda^*, \nu^*} \cdot \partial_\theta(\lambda^*, \nu^*) + \partial_\theta(\lambda^*, \nu^*; \theta) \\
\Rightarrow \partial_\theta(\lambda^*, \nu^*) &= -\left[\partial_{\lambda,\nu} \phi(\lambda, \nu; \theta)\right]^{-1}_{\lambda^*, \nu^*} \partial_\theta(\lambda^*, \nu^*; \theta)
\end{aligned}$$

In summary, this gives us a parametrization of bond-breaking as a constrained min-cut problem with a fast forward pass and a numerically stable backward pass.

Appendix D

Conditional denoising diffusion for structural elucidation

This section describes a proof-of-concept I carried out in the final weeks of my PhD for directly solving the MS structural elucidation inverse problem.

D.1 Background

Mass spectrometry structural elucidation is usually cast as an information retrieval problem: given a database of known (structure, spectrum) pairs – generated either experimentally or computationally – we identify an unknown spectrum by assigning it the identity of its closest match in the database. This faces a major limitation: we can only accurately identify a compound if it is present in our database. While it is in principle possible to produce an exhaustive database by enumerating feasible compounds, this is extremely inefficient as the space of compounds grows very quickly. ML approaches that directly solve the inverse problem by generating a molecule structure conditional on a mass spectrum have only very recently been developed [1, 2]. These represent molecules as SMILES strings, which are generated a character at a time autoregressively.

Recently, discrete denoising diffusion models have been developed and are effective for molecule generation [3]. We suggest a conditional form of this approach is particularly natural for structural elucidation. Firstly, modern tandem MS achieves sufficient mass resolution to permit inferring the precursor formula via mass decomposition with high accuracy [4]. In the context of discrete denoising diffusion, this specifies both the number of nodes in the graph, and their elemental composition: rather than generating graphs of arbitrary size, we need only generate an adjacency matrix of bonds (an *isomer*) compatible with the observed formula. (By comparison, it is unclear how to impose such a constraint when autoregressively generating a SMILES string.) We also believe iterative refinement of an adjacency matrix better matches the 'algorithmic structure' of structural elucidation than autoregressive generation of a SMILES string, and permits incorporating inductive biases about the relationships between the observed peaks and the estimated matrix in a straightforward manner. Importantly, here we show these inductive biases are necessary for effective structural elucidation and lead to large gains in performance.

D.2 Conditioning diffusion models on mass spectra

One obvious approach to implementing this diffusion model would be to train on spectrum-structure pairs, and at each step in the reverse diffusion process, concatenate the (un-noised, observed) mass spectrum as input to the model (e.g. a sequence of tokens) in addition to the current noisy estimate of the adjacency matrix. This is what we first attempted, but we saw limited success, both with real and synthetic data. In the case of real data (NIST-20 [5] spectra, labelled with formulas via SIRIUS [6]), it was unclear whether our model was actually using the spectrum at all in the generation process: valid isomers were being sampled, but they were no closer to the true structure than those sampled when dropping out the spectrum entirely. Ignorance of label information is a known issue in conditional denoising diffusion generally, but approaches we experimented with – including classifier-free guidance [7] – did not improve performance.

Switching to entirely synthetic spectra of simpler molecules improved the situation, but still (as we later show) underperformed. We conjectured an alternative explanation as to what was occurring: we supposed that spectrum information was being ignored because the conditional problem was too difficult to solve solely as iterative refinement of atom-to-atom links. In particular, we thought an important class of 'intermediate variables' were being omitted: the membership of atoms in peaks. This was in part inspired by work on graphically-structured diffusion models [8], which shows explicitly representing such variables can greatly boost performance in solving simpler combinatorial optimization problems. (Conveniently, these variables can themselves be treated as links to predict, in a heterogeneous graph comprising atom and peak nodes – and, unlike with real spectra, are observed and available as labels when using synthetic spectra.) If inferring whether a particular atom in the (current estimate of) the adjacency matrix belonged to a particular peak is a necessary subproblem to solve, by not explicitly representing this state, we are forcing the diffusion model to infer it during its forward pass – and moreover redoing that computation from scratch on every single forward pass. As early diffusion iterates do not specify useful adjacency matrices, this subproblem would likely not be solved very usefully either – making earlier iterates wasteful. There is also an information-theoretic view here: a discrete denoising diffusion of a symmetric adjacency matrix communicates at most $O(\binom{n}{2})$ bits across iterations. If it is necessary for information about peak-atom assignments to persist across iterations, there simply may not be enough space to do so.

Importantly, a useful byproduct of this approach would also be explicit association of a substructure with each peak, which has benefits for interpretability. It also might make the association between likelihood of a particular substructure compatible with a peak, and properties of its cut-set, more explicit to the model.

D.3 Methods

D.3.1 Data generation

Structures

To generate our synthetic training data, we begin by sampling small molecule structures from the PCMQM4M-v2 dataset [9]. We only include molecules of CHNOPSX elemental composition, and remove any molecules of nonzero charge or that have radical electrons. We additionally only include molecules with between 10 and 20 heavy atoms: this is just to allow fast testing of our proof-of-concept, which as described here entails multiple steps that scale quite poorly with molecule size. When multiple molecules share the same InChiKey connectivity substring, we discard all but one randomly. Of the molecules passing these criteria, finally we sample 100k at random.

Simulating mass spectra and fragmentation graphs

We use CFM-ID [10] to predict mass spectra and fragmentation graphs from these structures. Specifically we use `cfm-predict`, with default parameters for positive-mode ESI spectra, and retain only the top 30 peaks by height. This yields three spectra per structure, at low / medium / high energies. As `cfm-predict` only generates the final structures for each peak, and does not provide the fragmentation DAG, we also run `fraggraph-gen` (positive-mode, max depth 2), which generates protonated SMILES strings of all fragments and intermediates, and their graph connectivity. Both of these operations were run in parallel on the MIT Supercloud, which necessitated using a more recent version of RDKit than the one recommended by the CFM-ID authors. As that RDKit version uses a newer C++ standard, some minor modifications to the CFM-ID C++ code were also necessary.

D.3.2 Preprocessing

Our spectrum simulator represents substructures as SMILES strings, which only specifies substructures up to graph isomorphism. To supervise a denoising diffusion of atom-peak labels, it is necessary to generate such labels in the first place: i.e. we must pick a particular indexing of the nodes in the precursor molecule, and propagate it down the fragmentation graph by identifying isomorphic subgraphs. This raises a number of issues that we address here.

Steiner fragmentation subtrees

In a fragmentation DAG, there will generally be more than one path from the root node (precursor molecule) to any fragment. Our approach to avoiding conflicting labellings in such cases is to simply select a subtree from the DAG. As CFM-ID also models fragmentation intermediates, there are generally nodes in the DAG that do not have corresponding peaks, but are necessary to include in order to reach some peak from the precursor. Since label propagation (as we next explain) is an expensive operation, our strategy for selecting this subtree is, for each spectrum, to simply pick the smallest subtree that contains a path from

the precursor molecule to every observed peak (i.e. to at least one substructure with the same formula): this minimizes the number of propagations we must do. This can be cast as a rooted *directed Steiner tree* problem [11]: we take the `fraggraph-gen` fragmentation DAG and augment it with terminal nodes corresponding to each peak generated by `cfm-predict`, to each of which we add an inbound edge originating from any substructure with the same chemical formula.

We solve this exactly using Gurobi, assigning all edges a unit cost. We did not investigate the effect of different strategies for selecting this subtree – e.g. picking the subtree describing the most probable sequence of fragmentations (per CFM-ID transition probabilities) as opposed to the smallest. It is unclear how sensitive the final model should be to such choices (and we expect CFM-ID to be a more serious source of bias anyway) – we leave their investigation to future work.

Propagating atom labels

We can now assign an arbitrary atom indexing to the (protonated) precursor molecule, and propagate it to the fragment nodes via traversing the fragmentation subtree. We do so by solving an *approximate subgraph isomorphism* problem at each edge of this tree. (Approximation refers not to optimization – we solve exactly using Gurobi – but rather to the fact that CFM-ID permits certain rearrangement reactions, meaning generally a child will only be *close to* an edge subgraph of its parent, not exactly one.) We specifically find a matching matrix Π of nodes from parent to child that minimizes the bond-order discrepancy of matched bonds:

$$\begin{aligned} \min_{\Pi \in 2^{M \times N}} & \|\Pi A_G \Pi^\top - A_H\|_2^2 \\ \text{s.t. } & \Pi 1_N = 1_M \\ & \Pi^\top 1_M \leq 1_N \end{aligned}$$

where H is a child fragment of G on $M \leq N$ nodes respective, and the adjacencies A_G and A_H are weighted by bond order $\in \{0, 1, 2, 3\}$. (Currently, we actually minimize the 1-norm, and left-multiply the term inside the norm by Π. This is much faster, and appears to propagate valid labellings, but its correctness needs to be verified.)

We only run this optimization on the heavy-atom graph, which greatly boosts solution speed compared to including the hydrogens: we presently do not incorporate information on the hydrogen counts at all. (Importantly, this includes the special case of where the extra proton is located.) We leave tuning of this parameter for future work.

We also note we can conveniently store these indices directly in the SMILES string representation of each substructure using RDKit's 'atom map index' functionality.

Finally, there is a possibility that our Steiner tree includes (because it might minimize the overall cost of the tree) multiple substructures with the same formula: we have not checked how often this occurs, but in case it does, we sample a labelled substructure from all those compatible with each peak uniformly at random. (This might again benefit from incorporating CFM-ID's likelihood information).

D.3.3 Model

Molecule and spectrum featurization

An issue of some concern during the development of our model was GNN expressiveness: for example, we observed empirically that a Graphormer-like [12] architecture led to poor generation of ring structures. We additionally run into difficulties if we rely on feature engineering (e.g. ring information) to overcome GNN expressiveness issues, as the graph traversals involved are difficult to parallelize on GPU, and conversion back-and-forth between GPU and CPU graph objects at the start of each diffusion iteration is costly.

One approach we found effective (inspired by [13]) was to represent molecules as *complete graphs* with $\binom{N}{2}$ edges on N nodes, labelling edges where no bond is present as 'zero-order' bonds. To encode positional information, we compute Laplacian eigen-features of the molecular graph (to be clear, the sparsified version), using sign-flipping augmentation. Importantly these can be computed efficiently on the GPU, and contain information on graph connectivity and presence of cycles. (In particular we use all eigenvalues, not just the top few: the multiplicity of the zero eigenvalue is important here.)

To reduce the size of these graphs, we only use the heavy-atom subgraph, and represent hydrogens as a count feature per heavy atom. We therefore also must provide the formula of the precursor molecule as input to any model, as the noise process does not conserve the (always known) total number of hydrogens. (We omit the additional proton added by CFM-ID: the hydrogen counts represent those of a neutralized molecule. This may be worth revisiting: knowledge of where charge localizes is informative for this task.) We also strip stereochemistry information, and *kekulize* the molecules: aromatic bonds are converted to explicit single or double bonds. (This was done to eliminate a spurious correlation we observed, in which the model would label any ring structure as aromatic, irrespective of elemental composition or ring size.)

We additionally label each bond with an indicator of whether or not it participates in a cycle (which we later treat as an additional label to denoise). We suspect maintaining an explicit local indicator of this global structural feature is useful: in particular, it permits the model to 'decide' on the cyclicity of the molecule as a whole earlier on in the diffusion process (well before actually assembling its cycles). (Reassuringly, we found the model predicts this label with very high accuracy – but nonetheless believe an ablation study would be valuable here.)

We represent the spectrum as a set of (formula, height) tuples: formulas are fixed-length integer vectors of element counts. We square-root-transform the height (heights vary widely in order of magnitude; this lessens this effect, and can also be looked at as dividing by the MLE of standard deviation under a Poisson distribution), and normalize to a max height of 1. We augment these spectra in two ways: we drop out peaks with probability 10% (we expect redundant information between peaks on average), and we multiplicatively jitter peak heights by 10% (which we empirically found represents average height fluctation in NIST-20 between spectra of stereoisomers). We also point out each molecule is representead in our training data by *three* different spectra, one per each of the collision energies predicted by CFM-ID.

Diffusion process

We use the approach developed in DiGress [3]: at step t, our forward (noising) process preserves each label (bond type / bond cyclicity per atom-pair, number of hydrogens per atom, inclusion/exclusion of an atom-peak pair) with probability $1 - \alpha(t)$, and with probability $\alpha(t)$ independently resamples its value according to its marginal frequency in the training set. Our reverse (denoising) process takes as input a noisy estimate of these labels, plus the observed spectrum and precursor formula, and predicts probabilities on the simplex per label, from each of which a categorical sample is drawn independently. We use a cosine schedule for $\alpha(t)$, with 500 diffusion time-steps. (We wonder whether this is an appropriate schedule for our task – we observe minimal change past $\sim 2/3$rds of iterations – but leave tuning for future work.)

Model architecture

Our model is implemented as a pure transformer, in which each node and edge in the (densified) heterogeneous molecule-spectrum graph is represented as a token in a sequence. The specific inputs and outputs per token type are described in the following table.

Token	**Number**	**Input**	**Output**
Graph	1	MLP(precursor element counts) + MLP(eigenspectrum)	-
Atom	N	Element embedding + number-of-hydrogens embedding + Laplacian PE	Number of hydrogens $\in \Delta^5$
Atom-pair	$\binom{N}{2}$	Bond-type embedding + cyclicity embedding + summed Laplacian PEs of atoms	Bond type $\in \Delta^4$, cyclicity $\in [0, 1]$
Peak	K	MLP(fragment element counts) + Sinusoidal(height)	-
Atom-peak	NK	Membership embedding + MLP(fragment element counts) + atom Laplacian PE	Membership $\in [0, 1]$

Our efforts to ensure expressiveness pay a major price in speed: the attention cost here is *quartic* in the number of (atoms + peaks). This could potentially benefit from structured attention masking, as described by [8] – however the optimal conditional independence structure may require some experimentation. We additionally tried low-rank attention approximations [14], but found PyTorch dense attention (which is very well optimized) was faster for the small graphs we use.

The outputs indicated in the table are predicted as logits by linear output heads. We minimize an (unweighted) sum of cross-entropy losses for each type of label.

Model hyperparameters and training

The transformer uses a width of 256, depth of 8, and 8 attention heads. We use a batch size of 100, and Adam with a learning rate of 0.0003, training for 200 epochs using 4×2 GPU nodes on the MIT Supercloud. We did not carry out any hyperparameter tuning, beyond scaling each size parameter down by a factor of 0.75: the larger model gave lower validation loss.

We use a 95/5 formula-disjoint split of our data into training and validation sets. Investigation of more aggressive splitting strategies (e.g. scaffold / fingerprint splitting) is left to future work.

D.3.4 Ablation baselines

Our focus here is in determining how to solve structural elucidation via diffusion model; while we believe this has intrinsic advantages over previous approaches, we defer those comparisons to future work. Instead, we consider two alternative implementations of our diffusion model: one – the 'naive' approach – simply concatenates on the spectrum tokens to the molecule tokens, and does not incorporate the peak-to-atom labels at all. We also compare both approaches to performance of pure isomer sampling, without any MS information at all beyond the formula of the precursor: this permits us to tell whether the information in the spectrum is being used at all. (Otherwise all architectures and parameters are identical.)

D.4 Results and Discussion

D.4.1 Training curves

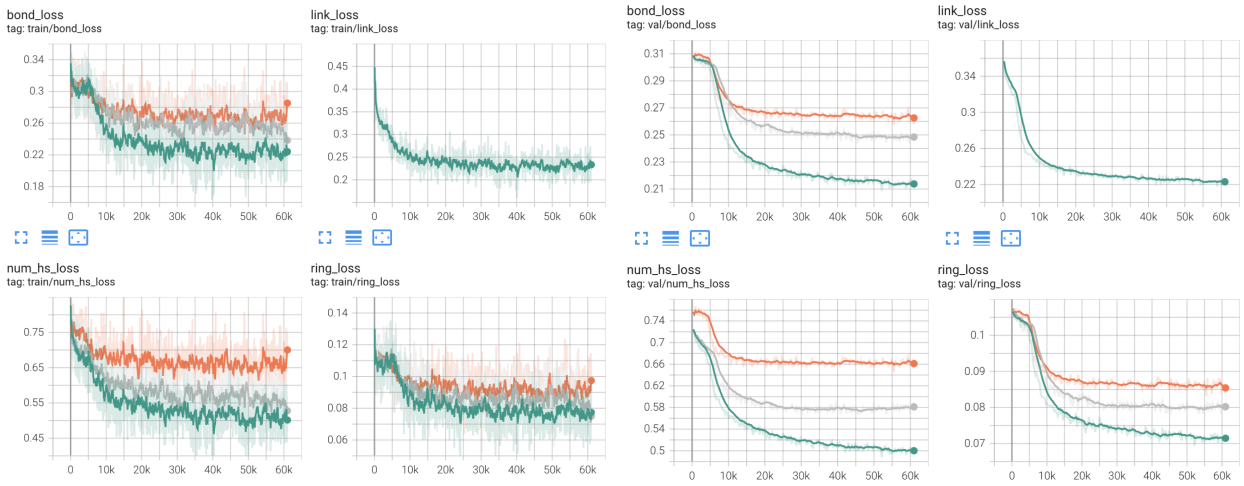

Figure D.1: Training and validation cross-entropies for our approach (green) and baselines (no peak-atom links: gray; no spectrum: orange).

Reassuringly, the models with richer denoising tasks achieve lower training and validation loss (i.e. predict greater probability for the correct labels); we also do not observe obvious

overfitting by 200 epochs. We suspect this evaluation is however not very meaningful: it might simply reflect the greater mutual information between inputs and outputs, as opposed to a more powerful model.

D.4.2 Structure elucidation accuracy

Our ultimate evaluation therefore carries out the *entire* reverse denoising process, starting with atom-atom and atom-peak adjacencies sampled from the prior. (This is what we would do at test time.) Because it is presently very costly to do this with our model, we only draw a *single* sample, for a single random collision energy from each of 1000 randomly-chosen structures in our validation set.

We evaluate the quality and accuracy of this sampled structure on average, checking for how frequently the sample describes a valid molecule (connectivity, valid valences); among these, how many yield the correct number of hydrogens (recalling we do not represent these explicitly); and how many yield an *exact* structure match. We also quantify structural similarity via Jaccard similarity (size of intersection over union of set bits) of RDKit molecular fingerprints of the sampled and true structure, as well as a bond edit distance between them (computed as the optimal value of the approximate isomorphism objective described above).

Model	Valid generated molecules	+ Correct hydrogen count	+ Exact structure match	Mean bond edit distance	Mean fingerprint Jaccard similarity
Atom-atom + peak-atom	879±10	**835±12**	**93±9**	**10.83±0.24**	**0.497±0.009**
Atom-atom only	**887±10**	823±12	20±4	14.36±0.24	0.417±0.007
No spectrum	859±11	110±10	1±1	23.45±0.26	0.243±0.004

Figure D.2: Structure elucidation performance of our approach and baselines. Standard errors of means are indicated.

All approaches sample valid molecules at approximately the same rate. However, purely marginal isomer sampling does much poorer at yielding molecules with the correct number of protons. This is surprising and not obviously an effect of excluding mass spectrum information. One possible explanation is that the MLP module we use to embed formulas is used for both precursor and fragment formulas: as the latter are omitted when excluding the mass spectrum, this subnetwork sees much less training data.

Importantly, we see training the diffusion model to explicitly generate peak-atom links helps greatly. It significantly boosts the means of both similarity metrics, and improves the exact structure match rate (when drawing a *single* sample) by almost 5×. Comparing to isomer sampling, which only guesses one structure correctly, we see both approaches to spectrum conditioning nonetheless do incorporate the spectrum information. This might reflect the simplicity of our training data – recalling that an early attempt (without peak-atom information) trained on NIST-20 spectra appeared not to use it.

The figure shows the Jaccard similarity versus bond edit distance. We suggest the observed bimodality in the Jaccard similarity is not 'real': it rather reflects a limitation of Jaccard similarity for evaluating the sorts of errors that occur in structure elucidation. When

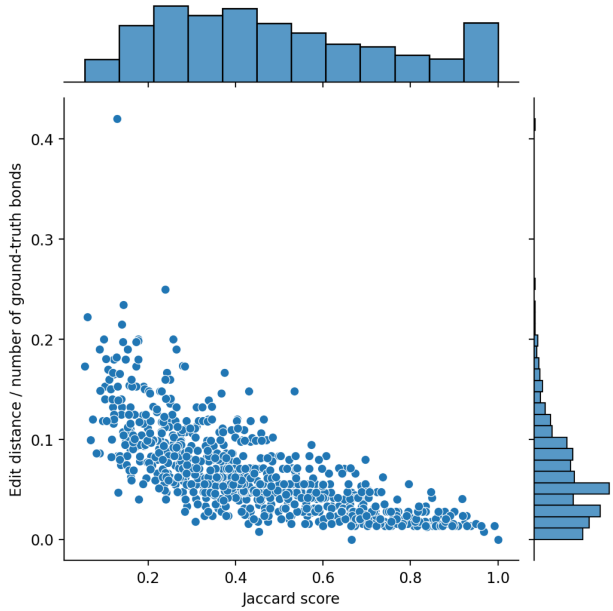

Figure D.3: Jaccard similarity versus bond edit distance for 1000 samples from our approach.

inspecting generations, a common failure mode we observe is the correct identification of subgraphs corresponding to spectral peaks, but their incorrect assembly (i.e. an incorrect arene substitution pattern). These errors correspond to small edits (such as changing the endpoint of one edge), and can often be difficult fundamentally to detect in mass spectra – however, they can lead to large changes in which graph motifs are present in the molecule, and therefore which particular bits are set in the fingerprint.

D.5 Conclusion

Here we demonstrate, using synthetic data, how discrete denoising diffusion might be adapted to solve the structure elucidation problem. In agreement with domain-agnostic work on using diffusion models to solve combinatorial optimization problems, we see that making an important class of intermediate variables – peak-atom assignments – explicitly represented in the diffusion process leads to major performance gains.

However, due to time constraints this work remains at the proof-of-concept stage: we anticipated substantial optimization of the approach will be necessary before it presents a practical algorithm for MS structural elucidation. Below we highlight some future directions.

- **Performance optimizations.** Both real small molecules and their mass spectra are typically larger than those used here. To scale to these, both the combinatorial routines we call during preprocessing, and our model architecture, will need to be better optimized (potentially using approximate solutions for the former; and for the latter, perhaps a more efficient architecture that maintains the desired expressivity, or requires fewer denoising iterates).

- **Hydrogen-count violations.** These are an unfortunate artifact of what is normally a sensible way to reduce the size of molecular ML problems. Rejection sampling may be a simple way to achieve this with a faster model.
- **Denoising fragmentation DAGs.** We point out there is a third class of edges we could potentially represent explicitly as well: peak-to-peak directed edges, representing the fragmentation DAG.
- **Better baselines.** This approach should be compared against the (handful of) existing ML methods for de-novo structural elucidation. We caution that the pretraining schemes employed by others will likely not translate readily to our context (nor ours to theirs), making fair comparison difficult.
- **Reranking predictions.** The evaluation described here looks merely at a single sample: we should ideally sample multiple structures per spectrum and rank them by some criterion – for example the similarity between the observed spectrum and a spectrum predicted from that structure by a separate forward model.
- **Real data.** It is unclear how to best apply this approach to real mass spectra, which are limited in number and for which the ground-truth substructures are generally unknown. One possibility would be to fine-tune a model trained using atom-atom and atom-peak labels on synthetic data (which would necessarily inherit the biases of its simulator) using solely atom-atom labels available in real data (omitting the atom-peak term from the objective entirely). Another option would be a hybrid approach, using CFM-ID or MAGMA [15] to label real spectra with synthetic substructures.

References

[1] Michael A. Stravs et al. "MSNovelist: de novo structure generation from mass spectra". In: *Nature Methods* 19.7 (May 2022), pp. 865–870. ISSN: 1548-7105. DOI: 10.1038/s41592-022-01486-3. URL: http://dx.doi.org/10.1038/s41592-022-01486-3.

[2] Thomas Butler et al. "MS2Mol: A transformer model for illuminating dark chemical space from mass spectra". In: (Sept. 2023). DOI: 10.26434/chemrxiv-2023-vsmpx-v4. URL: http://dx.doi.org/10.26434/chemrxiv-2023-vsmpx-v4.

[3] Clement Vignac et al. "DiGress: Discrete Denoising diffusion for graph generation". In: *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL: https://openreview.net/pdf?id=UaAD-Nu86WX.

[4] Oliver Fiehn. *Critical Assessment of Small Molecule Identification 2022*. http://www.casmi-contest.org/2022/index.shtml/. [Online; accessed 12-December-2022]. 2022.

[5] National Institute of Standards and Technology. *NIST-20*. 2020. URL: https://www.nist.gov/programs-projects/nist20-updates-nist-tandem-and-electron-ionization-spectral-libraries.

[6] Kai Duhrkop et al. “SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information”. In: *Nature Methods* 16.4 (Mar. 2019), pp. 299–302. ISSN: 1548-7105. DOI: 10.1038/s41592-019-0344-8. URL: http://dx.doi.org/10.1038/s41592-019-0344-8.

[7] Jonathan Ho and Tim Salimans. “Classifier-Free Diffusion Guidance”. In: *CoRR* abs/2207.12598 (2022). DOI: 10.48550/ARXIV.2207.12598. arXiv: 2207.12598. URL: https://doi.org/10.48550/arXiv.2207.12598.

[8] Christian Dietrich Weilbach, William Harvey, and Frank Wood. “Graphically Structured Diffusion Models”. In: *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*. Ed. by Andreas Krause et al. Vol. 202. Proceedings of Machine Learning Research. PMLR, 2023, pp. 36887–36909. URL: https://proceedings.mlr.press/v202/weilbach23a.html.

[9] Jin-Xian Hu et al. “Incorporating label correlations into deep neural networks to classify protein subcellular location patterns in immunohistochemistry images”. In: *Proteins: Structure, Function, and Bioinformatics* (Oct. 2021). DOI: 10.1002/prot.26244. URL: https://doi.org/10.1002/prot.26244.

[10] Fei Wang et al. “CFM-ID 4.0: More Accurate ESI-MS/MS Spectral Prediction and Compound Identification”. In: *Analytical Chemistry* 93.34 (Aug. 2021), pp. 11692–11700. DOI: 10.1021/acs.analchem.1c01465. URL: https://doi.org/10.1021/acs.analchem.1c01465.

[11] M. Hauptmann and M. Karpinski. *A Compendium on Steiner Tree Problems*. CS-Reports. Inst. fur Informatik, 2013. URL: https://books.google.com/books?id=09fpvgEACAAJ.

[12] Chengxuan Ying et al. “Do Transformers Really Perform Badly for Graph Representation?” In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc'Aurelio Ranzato et al. 2021, pp. 28877–28888. URL: https://arxiv.org/abs/2106.05234.

[13] Jinwoo Kim et al. “Pure Transformers are Powerful Graph Learners”. In: *NeurIPS*. 2022. URL: https://arxiv.org/abs/2207.02505.

[14] Krzysztof Marcin Choromanski et al. “Rethinking Attention with Performers”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL: https://openreview.net/forum?id=Ua6zuk0WRH.

[15] Lars Ridder, Justin J. J. van der Hooft, and Stefan Verhoeven. “Automatic Compound Annotation from Mass Spectrometry Data Using MAGMa”. In: *Mass Spectrometry* 3.Special$_{Issue_2}$ (2014), S0033–S0033. ISSN: 2187-137X. DOI: 10.5702/massspectrometry.s0033. URL: http://dx.doi.org/10.5702/massspectrometry.S0033.