Machine Learning Methods for Single Cell RNA-Sequencing Data to Improve Clinical Oncology

by

Rebecca Boiarsky

B.A., Yeshiva University (2014)
M.S., Columbia University (2016)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2025

Authored by:	Rebecca Boiarsky Department of Electrical Engineering and Computer Science May 16, 2025
Certified by:	David Sontag Professor of Electrical Engineering and Computer Science Thesis Supervisor
Certified by:	Gad Getz Core Institute Member of the Broad Institute of MIT and Harvard, and Professor of Pathology at Harvard Medical School Thesis Supervisor
Accepted by:	Leslie A. Kolodziejski Professor of Electrical Engineering and Computer Science Chair, Department Committee on Graduate Students

THESIS COMMITTEE

Thesis Supervisors

David Sontag
Professor
Electrical Engineering and Computer Science, MIT

Gad Getz
Core Institute Member
Broad Institute of MIT and Harvard

Thesis Readers

Marinka Zitnik
Assistant Professor
Biomedical Informatics, Harvard Medical School

Caroline Uhler
Professor
Electrical Engineering and Computer Science, MIT

Machine Learning Methods for Single Cell RNA-Sequencing Data to Improve Clinical Oncology

by

Rebecca Boiarsky

Submitted to the Department of Electrical Engineering and Computer Science
on May 16, 2025 in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

ABSTRACT

Single-cell RNA sequencing (scRNA-seq) offers a detailed view of the cellular and phenotypic composition of healthy and diseased tissues. While machine learning (ML) methods are well-suited for the high-dimensional nature of scRNA-seq data, current computational tools face limitations, particularly when confronted with data from clinical oncology. This thesis presents the development and application of ML techniques for scRNA-seq data to address key computational challenges, with a focus on challenges in clinical oncology. It covers four key areas: identifying gene signatures and biomarkers in multiple myeloma, developing methods to account for somatic copy number variations in tumor samples, benchmarking large, pre-trained scRNA-seq foundation models, and creating a framework for predicting clinical outcomes using patient-level representations of single-cell data. Together, these studies aim to develop and evaluate novel ML algorithms for scRNA-seq data which can unlock actionable insights for personalized medicine.

Thesis supervisor: David Sontag
Title: Professor of Electrical Engineering and Computer Science

Thesis supervisor: Gad Getz
Title: Core Institute Member of the Broad Institute of MIT and Harvard, and
Professor of Pathology at Harvard Medical School

Acknowledgments

I am first and foremost deeply grateful to my PhD advisors, David Sontag and Gaddy Getz. David initially hired me as a research software engineer prior to my transitioning to a PhD student in his lab. That role set me on a path that has changed my career forever, and I am so grateful that David believed in me and our shared vision to leverage machine learning for translational impact. David sets a very high bar for his students—in terms of scientific rigor, selecting impactful problems to tackle, and digging deep into the literature and methodology of previously published works—and at the same time, David cares deeply about his students as full people inside and outside of the lab. His seemingly endless energy and incisive insights inspired and motivated me to engage deeply with challenging, high-impact problems. From lab meetings, to our 1:1s, to lab hikes, to whiteboarding sessions, to going out of his way to meet my newborn babies, the imprint David has left on me will inspire me for years to come.

I joined Gaddy's lab at the end of my first year, in pursuit of a mentor and lab who could help guide us toward solving important, biological questions in oncology. I quickly learned that Gaddy was as kind as he was brilliant, and that he had nurtured a lab filled with people of the same character. Gaddy prioritized going deep instead of fast (but also going fast!), a value that can be easy to lose sight of in a competitive academic environment. He pushed my work to be more scientifically rigorous and reproducible, urging me to take time to do an analysis completely and correctly and convincing ourselves that our results were meaningful and reproducible before sharing them with the community. Zooming with Gaddy while he proofread every word of my manuscripts taught me the importance of communicating scientific results with clarity and candor.

Thank you to my thesis readers, Marinka Zitnik and Caroline Uhler, for your time giving feedback on my thesis over the past year. Both of your own research has and continues to inspire my own research direction, and I am humbled to have had your input and support.

I am grateful to the brilliant collaborators who worked on the works in this thesis with me. To Nick Haradhvala, my mentor in the Getz lab, for being the person by whom I wanted to run every idea and result, and for making time to meet with me regularly despite his jam-packed schedule and responsibilities. I'm not sure how you find time for everything, but thank you for all that you taught me over the past five years. Your insights improved almost every aspect of this work. To Nalini Singh and Alejandro Buendia, my partners in exploring and building single-cell foundation models. Our brainstorming about cell and patient-level foundation models represented to me like what a PhD is all about—understanding a new field deeply and thinking creatively about how to tackle the frontier of it. In addition to what we built together, your friendships made those years of my PhD so much fun. To Romain Lopez, my mentor during my summer internship at Genentech in 2023, as well as Jan-Christian

Hütter: I learned so much from your guidance on how to tackle a hard scientific problem in a short amount of time. Thank you to the brilliant Johann Wenckstern, who contributed and implemented core ideas to bring our work on patient-level foundation models over the finish line during his internship with us in 2024. I'd also like to acknowledge others who provided direct feedback on the works in this thesis: Ava Amini, Aviv Regev, François Aguet, Jean-Baptiste Alberge, Romanos Sklavenitis-Pistofidis, Irene Ghobrial, and Ming-Chieh Shih.

To all my friends in the Sontag lab, who made the PhD so much fun and helped MIT to feel like home. Over the years the lab was filled with more wonderful people than I can name here, but especially: Irene Chen, Monica Agrawal, Mike Oberst, Rahul Krishnan, Hunter Lang, Hussein Mozannar, Chandler Squires, Christina Ji, Zeshan Hussain, Ilker Demirel, Shannon Shen, Edward De Brouwer, and Fredrik Johansson. As well as the brilliant Master's students whom I had the privilege to mentor and learn from: Shannon Hwang, Rohan Kodialam, Justin Lim, and Sama Setty. To the Getz lab, which definitely has more people than I can name, but especially Danielle Firer, Mendy Miller, Julian Hess, Yifat Geffen, Jide Ezike, Michael Vinyard, Yo Akiyama, Serene King, Khrystofor Khokhlov, Arvind Ravi, and Petar Stojanov.

To the Technology Childcare Center in Stata, who lovingly cared for Ilan so that his Mommy could spend full days doing research. It's not every kid who can say their first school was MIT!

To my lifelong friends, and especially to Lauren and Cheli, my virtual co-workers throughout the pandemic and beyond. Zooming with you was the highlight of many days! Your friendship means the world to me.

Finally, to my family. To my parents, Rachel and Seth Peyser, for supporting me in my academic pursuits throughout my whole life and instilling in me the values of hard work, humility, and the importance of leading a life filled with purpose. Thank you for giving me every opportunity to invest in my education, starting from my early childhood, through college, my Master's, and my pursuit of this PhD. Your love and support has made me the person I am today; I would not be here without you. I love you, and I'm so grateful to have you as my parents. To Daniel, my love, who convinced me to expand my scientific horizons and pursue a PhD, who believed in me even at times when I didn't believe in myself, and most importantly, whose character enhances every aspect of my life. I'm inspired by the way you support me to pursue my dreams and aim higher, and the confidence you have that we can overcome challenges, like my moving to Boston to join MIT while we were dating! I would not be here without you, and I'm so grateful for your constant love, support, advice, and friendship. To the rest of my family: my sisters, my in-laws, my siblings-in-law: thank you for being my cheerleaders throughout this PhD and life. My grandparents, and especially my scientist grandfathers Ralph Nussbaum and Pincus Peyser, inspired me from a young age with their intellectual curiosity and love of learning, and they have always supported and been proud of my academic journey. And finally, what I am most proud to have produced during this PhD are my two beautiful children, Ilan and Ayla. Loving you and watching you grow has been my greatest joy. My greatest privilege in life is being your mother! You are constant reminders of the true purpose of my research, which is to leave the world a better, healthier place. I hope to deliver on this during my career as a PhD scientist.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Single-cell RNA-sequencing (scRNA-seq) has transformed our ability to study cellular heterogeneity, opening new avenues for understanding disease biology at unprecedented resolution. This technology has been especially illuminating in the field of oncology, where inter- and intra-tumor variability has been shown to be correlated with prognosis and treatment outcomes [1]. This thesis explores how machine learning (ML) can be harnessed to extract meaningful patterns from scRNA-seq data, with the ultimate aim of improving our ability to detect, interpret, and act upon complex signals in cancer.

1.1 A case study in multiple myeloma

In Chapter 2, we begin this thesis by deeply exploring single-cell RNA sequencing (scRNA-seq) data in the context of one specific cancer: multiple myeloma (MM). MM is a hematologic malignancy that originates from plasma cells—terminally differentiated B cells that reside in the bone marrow and are responsible for producing antibodies. The disease is characterized by the uncontrolled proliferation of malignant plasma cells, which can eventually crowd out healthy hematopoietic cells, disrupt normal immune function, and lead to a range of complications including anemia, bone lesions, renal dysfunction, and hypercalcemia. MM is notable for its stepwise clinical progression: it develops from asymptomatic precursor conditions such as monoclonal gammopathy of undetermined significance (MGUS) and smoldering multiple myeloma (SMM) before advancing to symptomatic, full-blown MM. While these precursor stages are clinically defined based on levels of monoclonal protein and plasma cell infiltration, the underlying biological events that drive progression are not fully understood [2–5].

Understanding the cellular and molecular dynamics that distinguish benign from malignant plasma cells is critical for improving early detection, risk stratification, and therapeutic

intervention. scRNA-seq offers a powerful window into this process, profiling gene expression at the level of individual cells. Unlike bulk RNA-seq, which averages signals across a mixed population of cells, scRNA-seq can disentangle heterogeneity within and across patient samples, capturing distinct cell states and subpopulations that may be obscured in aggregate data. This makes scRNA-seq an especially valuable tool in cancers like MM, where malignant and non-malignant plasma cells can coexist in the same bone marrow microenvironment and may exhibit subtle transcriptional differences at early stages of disease.

In this initial study, we analyze scRNA-seq data collected from bone marrow aspirates of patients representing different stages of disease progression, from MGUS to SMM and ultimately MM. Our objective is to identify gene signatures—that is, co-varying sets of genes that form transcriptional modules—that are associated with disease stage. Expression modules that correlate with disease stage may serve as biomarkers for prognosis, while those present in early-stage samples could be leveraged for early diagnosis or even as potential therapeutic targets. By identifying these gene programs, we hope to gain insight into the molecular mechanisms underpinning malignant transformation and to lay the groundwork for more precise risk stratification and intervention strategies.

As the bone marrow biopsies used for sequencing are made up of a mix of cell types, including both normal and abnormal plasma cells, one major benefit of having data at single-cell resolution is that we can distinguish abnormal and normal plasma cells within a patient sample in order to isolate the disease signal. The resulting clean separation allows us to explore the underlying biology of malignant transformation and progression with high resolution, removing signal from abnormal cells prior to downstream analyses.

After separating normal and abnormal plasma cells within each patient sample, we focus on characterizing the transcriptional profiles of abnormal cells across all disease stages. Our goal is to investigate how abnormal plasma cells differ from their normal counterparts, and how these differences evolve as the disease progresses. We also explore intratumor heterogeneity—that is, the diversity of cell states within a single patient's malignant plasma cell population. These insights are uniquely accessible through single-cell resolution data, which allows us to disentangle subtle yet biologically meaningful differences that would be masked in bulk analyses.

To uncover patterns of transcriptional changes across patients, we leverage automatic relevance determination non-negative matrix factorization (ARD-NMF) [6], a dimensionality reduction and unsupervised learning technique that decomposes the gene expression matrix into a set of additive, non-negative components. Each component can be interpreted as a gene expression module—a group of genes that are co-expressed across subsets of cells and may reflect shared biological programs. NMF is particularly well-suited to scRNA-seq data due to

its parts-based representation, which can help tease apart overlapping transcriptional programs without imposing orthogonality or negativity constraints that lack biological interpretability. By identifying shared NMF components across patients, we uncover gene expression modules that are not only consistent with known biology (e.g., immunoglobulin production, cell cycle regulation) but also highlight novel expression programs that may play a role in disease initiation and progression. Some modules are enriched in malignant cells across multiple patients and thus represent candidate markers of transformation, while others are more variably expressed, pointing to interpatient heterogeneity that may influence prognosis or treatment response.

This case study serves as both a proof of concept and a launching point for the methodological developments that follow. In the course of this analysis, we encounter several key challenges that are emblematic of broader issues in applying standard machine learning and statistical methods to cancer scRNA-seq data. These challenges include disentangling technical variation from biological signal, aligning expression programs across patients with varying disease states, learning a meaningful and interpretable representation of cells rather than working in high-dimensional gene-space, and making statistically-sound patient-level predictions, such as which patients will progress, using scRNA-seq data. These limitations underscore the need for new computational approaches that are more robust, interpretable, and biologically grounded. As such, the remainder of the thesis is dedicated to developing and evaluating such methods, building on insights and open questions that emerge from this initial exploration of MM.

1.2 Overcoming copy number effects in tumor cell embeddings

A prominent challenge that emerges early in our analysis of single-cell scRNA-seq data from MM patients in Chapter 2 is the fact that each patient's tumor is so transcriptionally distinct from others, that it is challenging to identify shared cell types, cell states, and gene modules across patients, even those with similar disease subtypes. We hypothesize that a major driver of the observed wide-scale transcriptional changes between tumors are somatic copy number variations (CNVs). CNVs are large-scale genomic alterations—such as duplications or deletions of chromosomal segments—that are a hallmark of many cancers, including MM. These structural changes can have profound downstream effects on gene expression, as gains or losses in DNA copy number often translate to corresponding shifts in the expression levels of affected genes. In scRNA-seq data, this manifests as broad expression differences across

large chromosomal regions, which can mask other meaningful biological signal that is shared across patients. While some of the transcriptional changes due to CNVs may be biologically significant in driving disease progression, others are simply present as mechanistic artifacts of the underlying genomic instability.

This presents a challenge for discovering disease-relevant gene signatures and for visualizing and clustering cells based on their disease activity, rather than simply according to the patient from which they originate. Because CNVs tend to be patient-specific and can span hundreds of genes, they can dominate the low-dimensional representations learned by standard algorithms, effectively masking subtler, but potentially more informative, biological signals. For example, two patients with similar malignant phenotypes but different CNV profiles might appear dissimilar in the learned latent space. As a result, the presence of CNVs can confound efforts to identify shared disease signatures, limiting the generalizability and interpretability of downstream analyses.

Chapter 3 addresses this issue by proposing a novel generative model that explicitly accounts for the impact of CNVs on gene expression when learning low-dimensional representations of cells. The key idea is to model gene expression as a combination of CNV-driven effects and residual biological variation, and to structure the latent space in a way that allows these effects to be disentangled. By doing so, the model can learn representations that are less biased by patient-specific CNVs and more reflective of transcriptional programs that are conserved across individuals or associated with disease progression.

This approach draws inspiration from both probabilistic modeling and recent advances in representation learning, combining elements of variational inference with domain-specific assumptions about the structure of cancer transcriptomes. Specifically, the model incorporates prior information about chromosomal architecture and CNV calling—either inferred directly from the scRNA-seq data or obtained from complementary assays such as whole exome sequencing—to guide the separation of copy number–related variation from other biological signals. This principled adjustment enables the downstream use of these representations in tasks such as clustering, trajectory inference, and differential expression analysis, with reduced risk of confounding by CNVs.

The development of this method is directly motivated by the limitations encountered in Chapter 2, where we saw first-hand how cells from different MM patients clustered separately from each other, despite the fact that our analysis revealed that patient tumors were made up of heterogeneous cell states, with certain cell states found in multiple patients, even those experiencing different stages of disease. We hypothesized that this is in part due to CNV effects obscuring shared cell-level disease biology across patients that scRNA-seq is uniquely positioned to reveal. More broadly, this chapter exemplifies a recurring theme

throughout the thesis: the need for machine learning tools that are specifically tailored to the complexities of cancer biology. Generic dimensionality reduction and differential gene expression techniques can be dominated by known sources of variation in cancer datasets, such as CNVs or aneuploidy, obscuring signal that might otherwise yield novel discoveries and insight. By embedding domain knowledge into model design, we can build more robust, interpretable, and biologically faithful tools for single-cell analysis.

Ultimately, the goal of this chapter is not only to mitigate technical artifacts, but to better capture the latent structure of cancer-related gene expression—structure that reflects meaningful differences in cell state, differentiation trajectory, or therapeutic response. In doing so, we lay the groundwork for more reliable cross-patient comparisons, deeper insights into tumor evolution, and improved identification of clinically relevant transcriptional programs. Chapter 3 thus represents both a direct response to early analytical obstacles and a concrete step forward in developing ML frameworks that are resilient to the high-dimensional, noisy, and heterogeneous nature of cancer single-cell data.

1.3 Foundation Models for Single-Cell RNA-Sequencing

In Chapter 4, we shift from incorporating biology-driven priors—such as those used in Chapter 3 to account for somatic copy number variation—toward evaluating models that instead rely on large-scale unlabeled data to learn cell representations in a self-supervised manner. These so-called foundation models for single-cell transcriptomics aim to capture generalizable structure across diverse datasets without requiring explicit modeling of biological or disease-specific context. Inspired by similar trends in natural language processing and computer vision, this chapter explores the promise and limitations of these models in the domain of single-cell genomics.

The central idea behind foundation models is that by learning rich, transferable representations during pre-training, they can serve as a universal backbone for many different applications—be it classification, generation, or inference. In the context of scRNA-seq, this paradigm has begun to take shape through models like scBERT [7], scGPT [8], Universal Cell Embeddings (UCE) [9], and other transformer-based architectures trained on large-scale single-cell datasets. These models aspire to capture high-level structure in gene expression data, analogous to how state-of-the-art transformer-based language models capture syntax and semantics in language [10, 11], thereby offering the promise of general-purpose embeddings that can be fine-tuned or directly applied across biological conditions, tissues, and technologies.

Chapter 4 critically evaluates this promise by benchmarking two single-cell foundation

models on the concrete task of cell type annotation, a cell-level task that was used for evaluation in both the scBERT and scGPT papers. While much of the recent literature on foundation models has focused on model scale, architecture, or pre-training corpus, there remains a gap in systematic, comparative evaluation: how do these models actually perform relative to simpler, more interpretable alternatives, especially in practical settings where labeled data may be sparse?

To answer this question, we compare foundation models against simple linear baselines, and perform ablation studies to probe which elements of the model's architecture and training technique contribute to downstream performance. Somewhat surprisingly, we find that simple linear models often match or even outperform foundation models in terms of accuracy, robustness, and data efficiency. This is particularly notable in few-shot scenarios, where only a handful of labeled examples are available for fine-tuning. In theory, foundation models should excel in such settings due to their pre-trained knowledge; in practice, however, we observe inconsistent gains from pre-training, both across different foundation models and across different biological datasets. Moreover, we report that the complexity of foundation models often comes with trade-offs, such as increased computational cost, longer training and inference times, and reduced interpretability, that may not always be justified by marginal improvements in performance.

Chapter 4 therefore serves two purposes within the broader thesis. First, it contributes to an empirical understanding of the value of transformer-based foundation models for representing single cells, informing our choice of how to represent cells in future analyses and models. We conclude that the benefits of foundation models over other, more biologically motivated cell representations such as scVI [12] or PCA embeddings are not yet clear. Second, it reflects the thesis's overarching commitment to developing machine learning methods that are not only theoretically ambitious but also practically useful and biologically grounded. By emphasizing careful benchmarking, methodological transparency, and the importance of baselines, this chapter cautions against the blind adoption of increasingly complex models without corresponding attention to context, data regime, and use case. It is our hope that such scrutiny will not only help guide responsible model selection but also inform future efforts to design foundation models that are better attuned to the unique challenges and opportunities of single-cell biology.

1.4 Patient-level Representation Learning from Single-Cell RNA-Sequencing

Building on the progress in representation learning techniques developed in the previous chapters, Chapter 5 introduces scSet, a novel transformer-based framework designed to bridge the gap between high-resolution single-cell data and patient-level clinical prediction tasks. This chapter addresses a central challenge in translational single-cell genomics: how to leverage the rich cellular detail captured by scRNA-seq to inform clinically meaningful decisions at the level of an individual patient. While many single-cell studies focus on understanding cell types, states, or developmental trajectories, fewer methods directly tackle the task of predicting patient-level outcomes such as disease subtype, treatment response, or survival—despite the growing importance of precision medicine in oncology and beyond.

A common approach to clinical prediction from scRNA-seq is to reduce the data via pseudobulking, in which gene expression counts are averaged or summed across all cells in a patient sample. While pseudobulk strategies can simplify the modeling pipeline and make use of existing bulk RNA-seq tools, they do so at the cost of discarding valuable information about cellular composition, intercellular variation, and rare cell populations—features that are often critical for understanding disease heterogeneity and predicting outcomes.

scSet departs from this paradigm by treating each patient sample as a set of cells and learning to encode these sets directly into informative, low-dimensional representations using a permutation-invariant deep learning architecture. Specifically, scSet adapts ideas from the transformer family of models, which have proven highly successful in language modeling, vision, and protein structure prediction [10, 13–17]. The use of self-attention enables the model to flexibly capture interactions among cells, weigh their relative importance, and aggregate information in a data-driven way. This architecture respects the unordered nature of cellular data and can accommodate varying numbers of cells per patient, as opposed to many other ML architectures which require fixed input sizes or order.

The result is a learned embedding space in which patient samples are positioned according to their underlying transcriptional profiles, cellular heterogeneity, and inferred biological signals. These embeddings can then be used for downstream tasks such as classification (e.g., predicting whether a patient will respond to immunotherapy), regression (e.g., estimating time to progression), or clustering (e.g., identifying subgroups with shared biology).

In addition to designing and validating the scSet architecture, Chapter 5 explores the benefits of pretraining on large, unlabeled single-cell datasets prior to fine-tuning on smaller, labeled clinical cohorts. This follows the broader trend in machine learning toward transfer

learning and semi-supervised approaches, particularly valuable in the clinical domain where labeled data can be scarce or expensive to acquire. We develop a diffusion-based decoder that generates individual cells conditioned on a learned patient-level representation. By stacking our encoder together with this decoder, and training them jointly in an end-to-end autoencoding framework, we learn meaningful patient representations without requiring any labeled data. We find that pretraining scSet on large, diverse scRNA-seq datasets improves performance on clinically relevant tasks in smaller cohorts, suggesting a path forward for building generalizable models that can adapt to specific patient populations with minimal supervision.

Taken together, this chapter demonstrates how modern representation learning techniques, when thoughtfully adapted to the structure and scale of single-cell data, can enable clinically actionable predictions from data that was previously too high-dimensional, noisy, or heterogeneous to interpret at the patient level. scSet thus represents a step toward fulfilling the promise of single-cell genomics in precision medicine—where molecular insights gleaned from a patient's own cells can inform diagnosis, prognosis, and treatment selection. More broadly, this work illustrates how advances in machine learning architectures can be harnessed not only to model cells, but to serve patients.

Taken together, the studies in this thesis are unified by a central goal: to develop and evaluate ML methods that unlock the full potential of scRNA-seq data for understanding and treating disease. From uncovering early biomarkers in MM, to leveraging biological priors to learn better tumor cell representations, to benchmarking cutting-edge models for single-cell representation learning, to enabling actionable clinical prediction, each chapter addresses a distinct study in ML for computational oncology.

Chapter 2

Identifying Genes and Gene Modules to Characterize Precursor Stages of MM

2.1 Introduction

Multiple myeloma (MM) is a plasma cell (PC) malignancy residing in the bone marrow (BM) [2]. MM is almost always preceded by the precursor states monoclonal gammopathy of undetermined significance (MGUS) and smoldering multiple myeloma (SMM) [2–4]. However, the progression risk is highly heterogeneous, whereby certain patients progress quickly, while others never do. Patients with SMM exhibit progression rates of 10% per year, compared to just 1% for MGUS [18, 19]. Currently, our ability to predict progression is mostly based on a few clinical parameters (e.g., M-spike, light chains, and percent tumor burden) [20, 21]. Therefore, there is a need to further define molecular characteristics of patients who are at risk of progression. A thorough characterization of precursor cells and the state of the microenvironment in MGUS and SMM patients can help us distinguish the molecular mechanisms that underlie initial tumorigenesis versus later progression, predict which individuals are most at risk for progression, and identify potential targets for early therapeutic intervention.

Our understanding of genetic changes associated with disease progression and tumor evolution in MM is founded on studies that use clinical laboratory results and bulk analysis, including microarrays and DNA sequencing [5]. It has been shown that MGUS and SMM clones already harbor chromosomal alterations that define MM (translocations involving IgH or hyperdiploidy) [3, 21], and that progression to MM may be driven by the acquisition of secondary genetic events and mutational processes [22, 23]. As such, stratification of SMM patients was recently updated by the International Myeloma Working Group to include

cytogenetic abnormalities [24]. Additional studies showed that integration of events including *MYC* rearrangements, *TP53* mono- and bi-allelic inactivation, and *RAS* mutants indeed help stratify patients into novel models of progression to active MM [25, 26].

At the RNA level, it is challenging to draw conclusions about the phenotype of precursor cells and the dynamics of malignant transformation from bulk RNA-sequencing studies [27] due to low tumor purity (i.e., fraction of tumor cells in a sample) at the precursor stages. Recently, single cell studies of precursor conditions [28, 29] have allowed for characterization of these cells, but such datasets are still scarce and require careful computational analysis in order to glean insights from the limited number of abnormal cells present in biopsies from patients with precursor disease.

In this chapter, we share results previously published as part of Boiarsky et al. [30], in which we generated and analyzed single cell RNA-sequencing (scRNA-seq) data from 29,387 PCs representing 26 samples from patients with MGUS, SMM, or MM as well as 9 normal bone marrow donors (NBM). The single cell resolution of our data and the fact that those precursor samples contain a mixture of abnormal and normal cells allowed us to isolate the transcriptional changes of abnormal cells compared to normal plasma cells within the same patient sample and to characterize their transcriptomes across the disease spectrum. A related study previously analyzed the immune microenvironment of these same patients [31] (Table A.1), and here we explore transcriptional changes within tumor cells as well as correlations between tumor and immune cell activity in our cohort. We employed novel methods for identifying abnormal cells from within a mixed sample, report our findings from a nuanced within-patient differential expression analysis approach, and employ automatic relevance determination non-negative matrix factorization (ARD-NMF) [6] to highlight gene signatures that are active in our cohort and validated in external cohorts. Taken together, our study (i) presents a highly detailed and comprehensive view of the transcriptional transformation occurring in individual patients with myeloma and its precursor conditions, (ii) discovers gene expression signatures that are shared across patients with different driver events and at different stages of disease, and (iii) characterizes heterogeneity both between and within tumors.

2.2 Results

2.2.1 Single cell transcriptional profiles reflect driver events and reveal patient-specific patterns

To investigate the gene expression dynamics of PCs at different stages of MM progression, we performed droplet-based single-cell RNA sequencing of 35 samples isolated from BM aspirates of patients with MGUS (n=6), SMM (n=12), newly diagnosed MM (n=8), and nine healthy donors (NBM, n=9; Figure 2.1 a; Table A.2, Table A.3). One patient was biopsied both at the SMM stage and after progression to MM (SMM-1 and MM-8). Patients were followed for a median of 5.26 years (1921 days; range[1400, 5314]). Of the 18 patients with MGUS or SMM, 0/6 MGUS and 7/12 SMM patients were observed to progress to MM (Table A.2). After filtering cells using standard quality controls, we analyzed a total of 29,387 single CD138+ PCs (∼850 from MGUS, ∼8.4×103 from SMM, ∼1.7×103 from MM, and ∼9×103 from NBM). The number of CD138+ cells analyzed per sample ranged from 40 to 3,414, with a median of 591 (Table A.3). Projecting cells onto a 2D Uniform Manifold Approximation and Projection (UMAP) plot, we observed that cells from our NBM samples grouped together, while the majority of cells from patients with precursor conditions and overt MM formed separate groups of cells (Figure 2.1b,c).

Applying Leiden clustering [32], we obtained 25 clusters of cells (cell cluster assignments were published as Supplementary Data in [30]). Seven of these clusters represented healthy cells as determined by the majority of cells in these clusters coming from NBM samples and their overexpression of genes such as CD27. We merged these clusters into one "healthy" cluster (Figure 2.1d). Of the remaining 18 clusters, 11 each consist almost exclusively of cells from a single sample, reflecting the fact that normal variation between healthy individuals was minor compared to disease-associated expression changes (Figure 2.1e). With a few exceptions, the clusters that represented multiple samples grouped cells with shared disease biology: cluster 12 contained two sequential samples from the same patient, cluster 21 contained proliferating abnormal cells from 15 patients across disease stages, and clusters 3 and 20 (together with cluster 24) represented all patients with a t(11;14) translocation (MGUS-6, SMM-4,6,9,12; Figure 2.2). We further confirmed that batch effects such as age, sex, and sample preparation were not driving clustering results (Figure 2.3).

Of note, our cohort included one patient, SMM-12, whose biopsy included two distinct subclones. Both subclones harbored a t(11;14) translocation and clonally expressed IgG kappa, suggesting that they descend from the same parental clone, but only one acquired a CD20+ phenotype (Figure 2.4a,b), a MM phenotype occurring in up to 22% of patients [33].

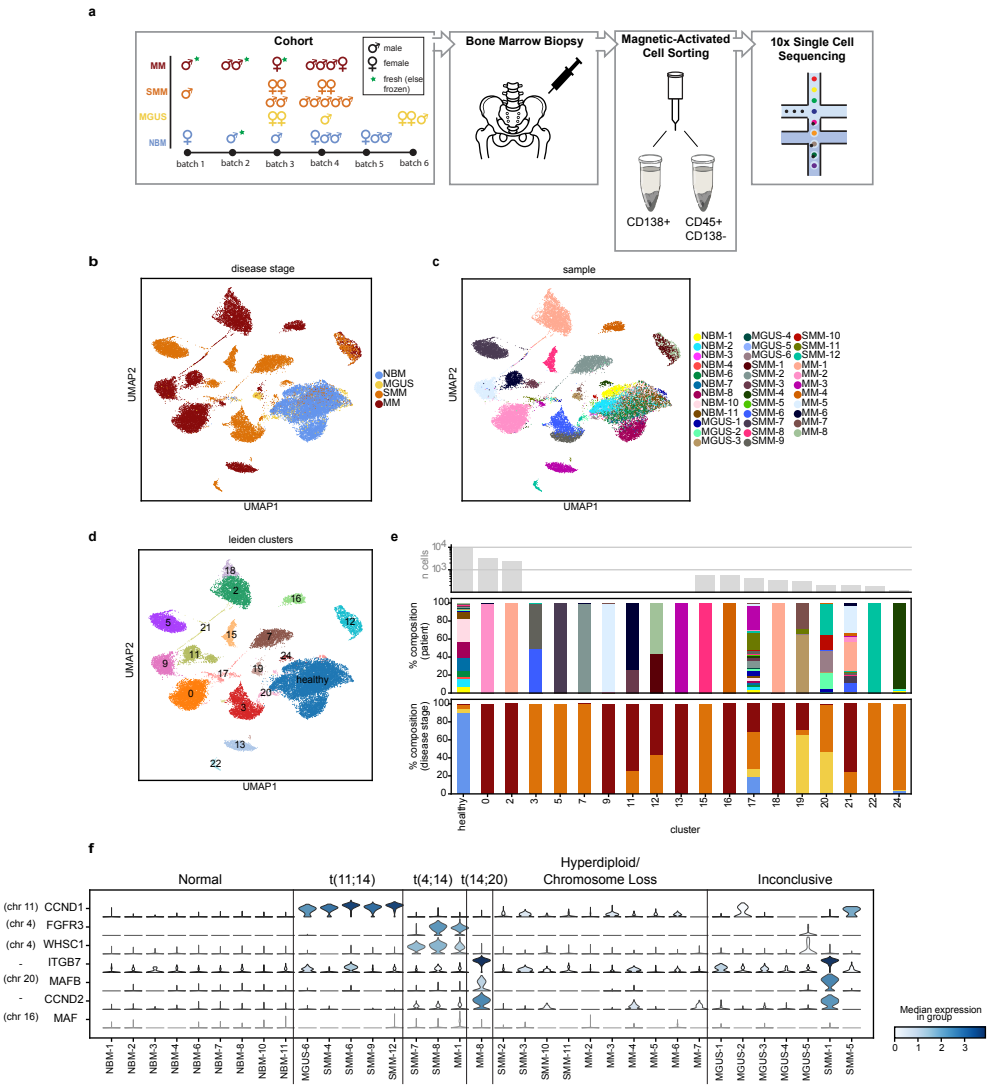


Figure 2.1: The landscape of healthy and abnormal plasma cells at single cell resolution. a, Overview of cohort and experimental setup, including the number of samples per disease stage, sex, sample preparation batch, and whether the sample was fresh or stored frozen prior to 10x sequencing. CD138+ bone marrow cell fractions were isolated and are analyzed in this study. a,b, UMAP representation of plasma cells colored by disease stage (b) and sample ID (c). Cells similar in expression profile are placed nearby in this embedding. d, Results of leiden clustering of all cells. Seven clusters were merged to define a single cluster of normal plasma cells. e, Sample composition of leiden clusters, by disease stage and sample ID (colors match the legends given in (b) and (c), respectively). The majority of clusters each consist of cells from a single sample. f, Violin plots showing distribution of expression of genes commonly upregulated in patients with translocations (y-axis), along with annotations of the cytogenetic alterations detected in samples by clinical iFISH assay (top).

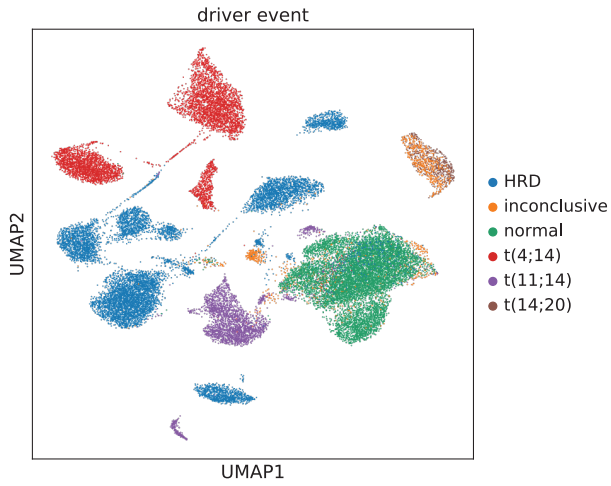


Figure 2.2: UMAP representation of plasma cells colored by cancer driver event, as determined by iFISH. Driver event is one of: normal, translocation (type specified in legend), hyperdiploid (HRD), or inconclusive test results.

This falls in line with previous studies that have shown CD20+ myeloma cells to be correlated with translocation t(11;14) [34]. Cells from the CD20- subclone clustered together with cells from other t(11;14) samples in cluster 20, while cells from the CD20+ subclone clustered separately in cluster 22, suggesting large expression changes associated with CD20+. Indeed, comparing gene expression in the CD20+ vs. CD20- subclones, we found 455 differentially expressed genes (DEG) ($|\log(\text{fold change})| > \log(1.5)$; false discovery rate $q<0.1$), with the top DEGs by q-value reflecting the B cell-like phenotype of these cells, including overexpression of *CD74*, *CD20* (also known as *MS4A1*), and HLA class II genes such as *HLA-DRA* and *HLA-DRB1* (Figure 2.4c; full DEG lists were published as Supplementary Data in [30]).

To benchmark the performance of our experiment, we used interphase fluorescence in situ hybridization (iFISH) to identify large-scale structural genomic variants (Table A.2) and then inspected the expression levels of translocation target genes cyclin D1 (CCND1), MM SET domain (MMSET/WHSC1), fibroblast growth factor receptor 3 (FGFR3), MAF BZIP transcription factor (MAF), and MAF BZIP transcription factor B (MAFB), as well as cyclin D2 (CCND2) and Integrin Subunit Beta 7 (ITGB7), whose overexpression is also associated with translocations. MM cells from patients with iFISH-reported translocations exhibited overexpression of the respective target genes (Figure 2.1f). In 4/7 samples whose iFISH results were inconclusive due to insufficient cell numbers, we were able to observe the overexpression of translocation partner genes or CCND2 and ITGB7, indicating possible corresponding translocations. Our patient with sequential samples at SMM (SMM-1) and after progression to MM (MM-8) confirms the ability of RNA-sequencing to capture the cytogenetic phenotype even prior to iFISH; while this patient's iFISH results were inconclusive

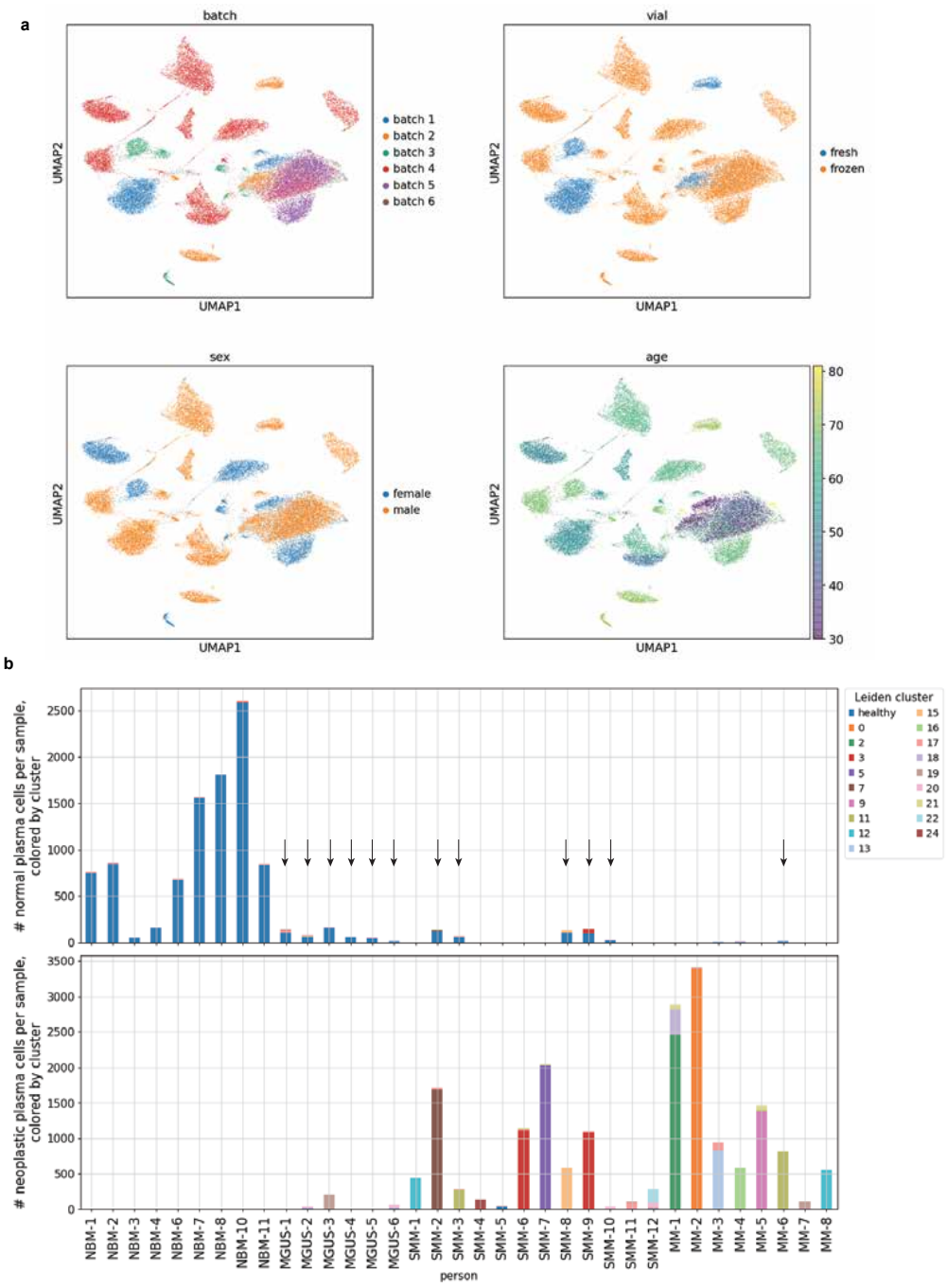


Figure 2.3: a, UMAP plot of all CD138+ cells colored by sample preparation batch, vial (whether the cells were fresh or frozen prior to sequencing), sex, and age. b, The cluster assignments for cells from each sample, separately for normal cells (top) and abnormal cells (bottom), with normal/abnormal status determined using the per-sample clustering technique. Arrows point to normal cells from patients that cluster together with normal cells from NBM, rather than together with the other cells from the same patient donor. This pattern suggests that the disease signal's influence on clustering is stronger than that of a potential batch effect.

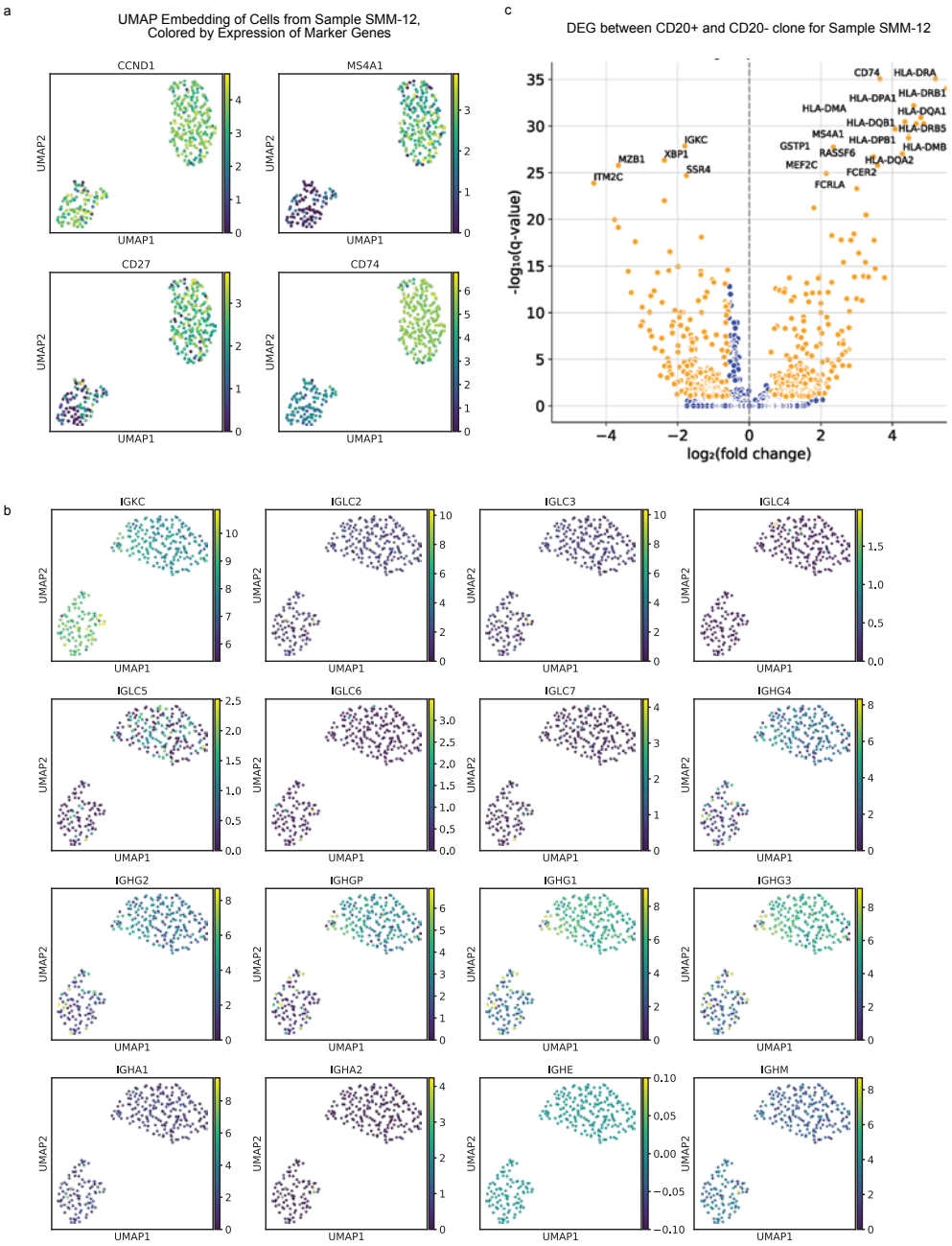


Figure 2.4: a, UMAP embedding of cells from sample SMM-12, which has 190 cells belonging to a CD20+ subclone (cluster on right) out of 284 total cells. Both clusters express CCND1, signifying that both harbor a t(11;14) translocation, but the cluster on the right expresses higher levels of CD20 (also known as MS4A1), CD27, CD74. b, Volcano plot of DEGs between the CD20+ and CD20- subclones from sample SMM-12. Orange denotes a significant DEG ($|\log(\text{fold change})| > \log(1.5)$; $q<0.1$). The top genes, ranked by q-value, are annotated on the plot. c, UMAP embedding of cells from sample SMM-12, colored by expression of immunoglobulin constant region genes. Both subpopulations express a kappa light chain (encoded by the gene IGKC) and IgG heavy chain (encoded by genes named IGHG*).

in the sample taken during SMM, we were able to detect overexpression of ITGB7, MAFB, and CCND2 at the transcriptional level, suggesting a t(14;20) translocation, which was later confirmed by iFISH after the patient's progression to MM.

2.2.2 In silico dissection of normal and abnormal cells within samples allows for characterization of disease even in samples with low tumor purity

One major benefit of studying precursor disease at single cell resolution is the ability to separate normal and abnormal CD138+ cells within each sample prior to downstream analyses. No individual marker genes can reliably distinguish these populations, but our full-transcriptome data enabled aggregation of an abnormal signal across many genes. To this end, we clustered the cells from each individual sample based on its highly variable genes (but excluding genes located in immunoglobulin loci), and then examined patterns of immunoglobulin and MM driver gene expression in each cluster in order to label the cluster as containing normal or abnormal cells (Figure 2.5; methodological details included in Section 2.4.8,). To complement and validate this method, we also developed a Bayesian hierarchical model for estimating the tumor purity of each individual sample based only on the distribution of immunoglobulin light chain expression (Figure 2.25, Figure 2.26; methodological details included in Section 2.4.7). Comparing these results, we observed strong agreement between the two purity estimation methods (Figure 2.6a). Our labels closely matched the Leiden clustering results (though not identically, highlighting the benefit of our curated labels), with 97% of cells we labeled as normal and <1% of cells we labeled as abnormal belonging to the healthy Leiden cluster (Figure 2.6b).

Our purity results suggest that samples from patients with precursor conditions have a sizable fraction of healthy PCs, as expected. On average, MGUS samples contained 73% healthy cells and SMM samples contained 8%, compared to just 0.5% in MM. Furthermore, the variability of tumor purity values was also greater at early stages of disease. Whereas MM samples had consistently high tumor purity (range 0.98-1), we observed increasingly large variability in SMM (0.58-1) and MGUS (0-0.81), respectively. For our downstream analyses, we separated normal and abnormal cells within each sample and characterized them independently.

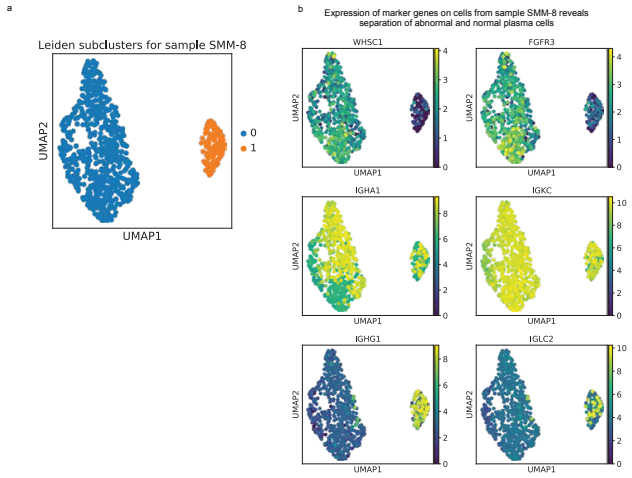


Figure 2.5: a, UMAP plot showing clustering for representative sample SMM-8. Genes in immunoglobulin loci were not used in the computation of the UMAP embedding or Leiden clustering. b, Expression of genes used to determine that cluster 0 contains abnormal cells and the cluster 1 contains normal cells: t(4;14) translocation associated genes WHSC1 and FGFR3 (first row); genes encoding the IgA-kappa immunoglobulin expressed on this patient's monoclonal cells (second row); genes encoding non-clonal immunoglobulin components IgG (IGHG1) and the lambda light chain (IGLC2) (third row). We observe that cells in cluster 0 clonally express IgA-kappa genes, while cluster 1 contains a mixture of cells expressing IgA and IgG heavy chains and kappa and lambda light chains. We performed a similar analysis for every patient sample in our cohort.

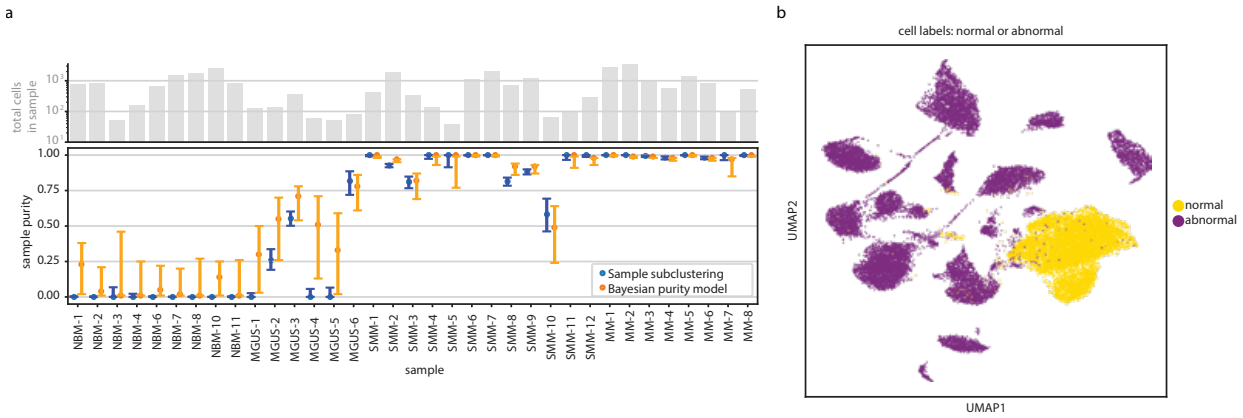


Figure 2.6: Identifying normal and abnormal plasma cells within patient samples. a, The number of cells (top) and estimated purity of each sample with 95% confidence intervals (bottom). Sample purity was estimated using two orthogonal methods: clustering of individual samples (blue; the fraction of cells labeled abnormal per sample is plotted) and our Bayesian hierarchical purity model (orange; the mode of posterior sample purity is plotted). b, UMAP localization of individual cells labeled normal or abnormal.

2.2.3 Transcriptional differences between abnormal and normal cells across patients

We performed a differential expression (DE) analysis comparing abnormal and normal cells. To this end, we split samples into their abnormal and normal populations, which we refer to as "pseudosamples." We compared abnormal pseudosamples to normal pseudosamples using limma-voom [35, 36] and found 764 DEGs ($|\log(\text{fold change})| > \log(1.5)$; false discovery rate $q<0.1$; full DE results were published as Supplementary Data in [30])).

In addition to genes known to be important for MM biology like *CCND1* [5] (upregulated), *CD27* [37–39] (downregulated) and *TMSB4X* [40] (downregulated), we also found other strongly regulated genes whose connection to myeloma is less well characterized. The top 4 upregulated DEGs by q-value included *RBFOX2*, *STIM1*, a transmembrane protein that mediates store-operated calcium entry and is of interest in multiple cancers [41–43], *IFIT1*, and the long non-coding RNA RP11-395G23.3. The top 4 downregulated genes included *CD19*, a B cell lineage marker gene which has been explored as a therapeutic target in MM despite its low expression [44, 45], *CTSH*, *CD81*, which regulates *CD19* [46] and has been shown by flow cytometry to be downregulated in MM and precursor conditions [47], and *ITGB2*. We also observed upregulation of *PSMB4* and *HSPB1*, which are associated with the proteasome (Figure 2.8a).

Unsupervised clustering of the pseudosamples based on their expression of these 764 DEGs showed good separation of abnormal and normal samples, as expected, and also revealed that hyperdiploid patients exhibit especially high expression of the upregulated DEGs and tend to cluster together. Abnormal samples do not cluster by disease stage, underscoring the fact that many of these DEGs are altered in both myeloma and precursor samples. All abnormal populations from SMM samples clustered together with the myeloma cells, while from MGUS, one abnormal pseudosample clustered with myeloma cells and two clustered with the normal cells. The two abnormal samples that cluster together with normal cells came from MGUS patients with very low numbers of abnormal cells detected (n=35 and 67 cells for MGUS-2 and MGUS-6, respectively). Thus we could not conclude that the MGUS phenotype is similar to that of normal cells, since the averaged gene expression in those pseudosamples is inherently noisy. Abnormal cells from MGUS-3 (n=205), on the other hand, clustered together with other abnormal samples (Figure 2.8b).

Interrogating MSigDB hallmark genesets, we found that pathways related to E2F targets, Notch signaling, G2M checkpoints, interferon alpha response, and Wnt/beta-catenin signaling are differentially enriched in abnormal samples compared to normal (t-test $q<0.1$; Figure 2.10). When comparing pathway enrichment results between MGUS and normal samples, we found

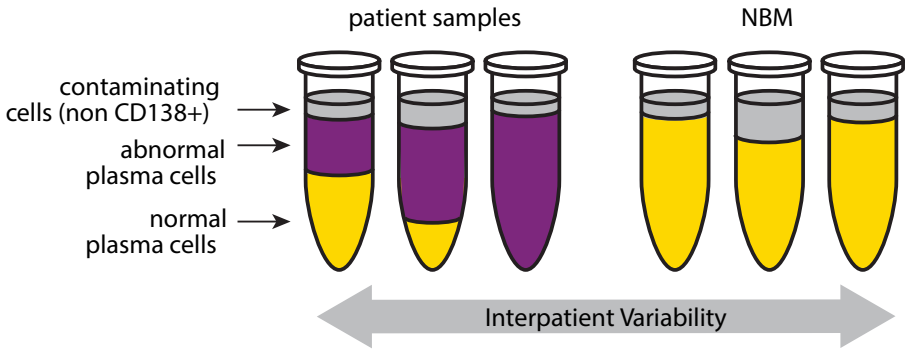


Figure 2.7: Cartoon schematic of our differential expression analysis. We run two DE analyses: First, we compare all abnormal (purple) vs. all normal (yellow) cells using limma-voom (Figure 2.8). Next, we compare patients' abnormal cells to their own healthy cells, controlling for inter-patient variability. Samples with 100% normal or abnormal cells were excluded from the within-patient analysis (Figure 2.9).

that the Wnt/beta-catenin pathway is already upregulated (t-test q=0.08). Individual upregulated genes from the Wnt/beta-catenin pathway include *DKK1*, *KAT2A*, and *TP53* (limma-voom abnormal vs. normal q<0.025). Low sample size (n=3) for abnormal MGUS samples may have hindered our power to discover other pathways that are already differentially enriched in MGUS vs. normal.

This DE analysis provides a general view of genes whose expression is consistently altered in disease, but it does not allow us to discover genes whose expression may be altered in just a small subset of patients in our cohort. Additionally, while normal cells are more similar to each other than abnormal cells are (Figure 2.1b,c), inter-patient differences still exist among them (Figure 2.11). Thus, this analysis suffers from both high variance due to the small number of normal samples and confounding effects due to non-disease-related differences between individuals that contributed healthy bone marrow and tumor samples. We address these limitations with the following analysis.

2.2.4 Within-patient abnormal vs. normal cell comparisons highlight inter-patient heterogeneity and patient-specific disease characteristics

To account for the limitations of the DE analysis described above, we leveraged samples containing mixtures of normal and abnormal plasma cells to perform a "within-patient" characterization of the disease. For each patient, we compared their abnormal plasma cells to their own healthy plasma cells (Figure 2.7). This allowed us to specifically characterize the

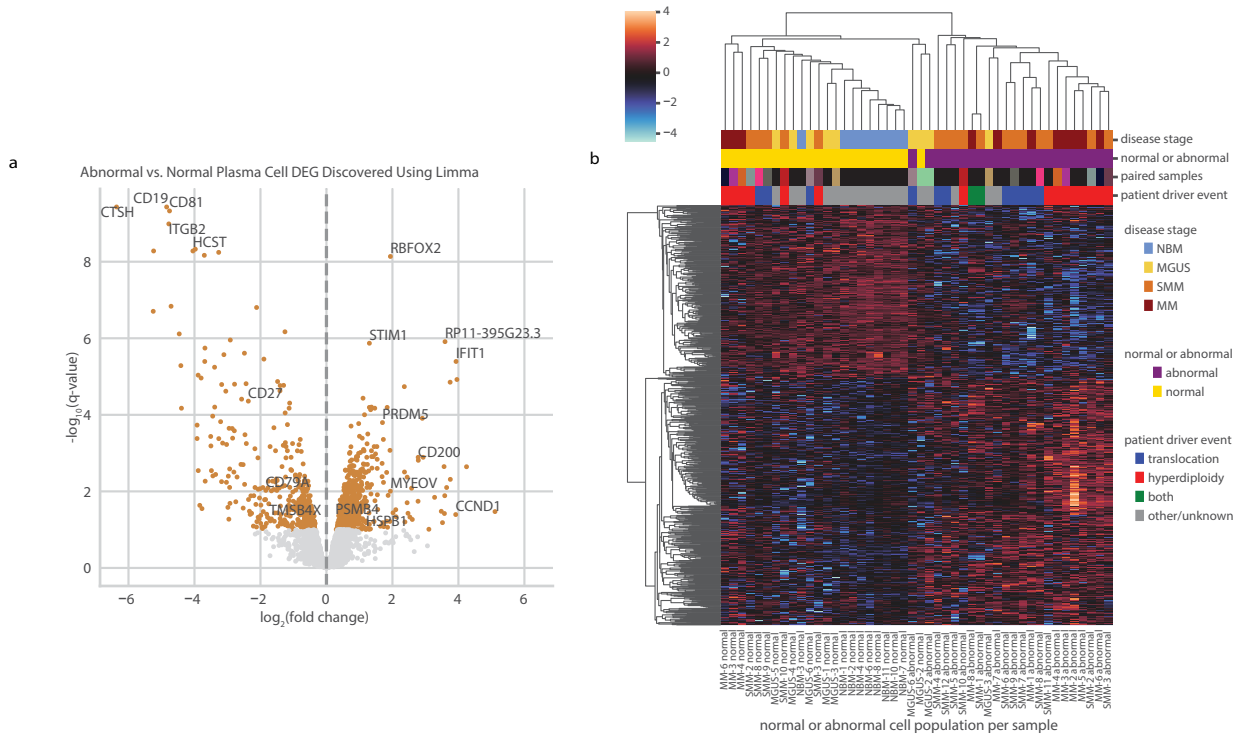


Figure 2.8: a, Volcano plot of limma-voom DE results for abnormal vs. normal cell populations. Orange denotes genes with q-value$<$0.1. The 4 most significantly up- and downregulated genes and other selected genes are annotated. b, Pseudobulk expression of DEGs detected between abnormal and normal pseudosamples using limma-voom (z-scored per gene). Each column represents the normal or abnormal cells from a given sample. Color annotations denote disease stage (top), normal or abnormal (second), paired columns coming from the same patient (third; matching colors denote that columns correspond to the same sample; black denotes that there was no paired sample), and whether IgH translocation or hyperdiploidy was detected in that sample by iFISH (bottom).

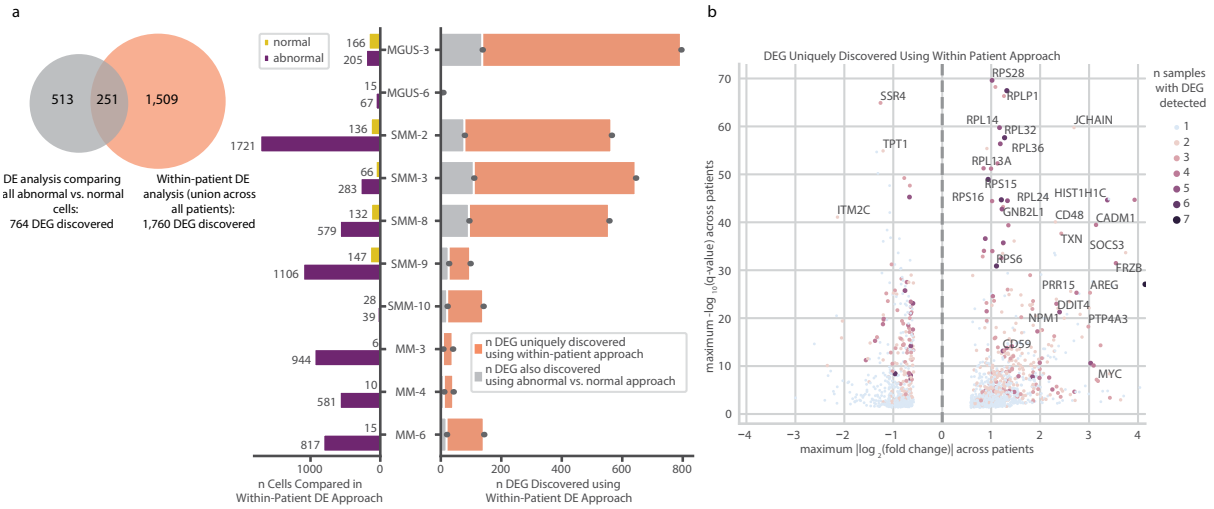


Figure 2.9: In silico dissection of transcriptional differences in normal and abnormal plasma cells within patient samples. We compare patients' abnormal cells to their own healthy cells, controlling for inter-patient variability. Samples with 100% normal or abnormal cells were excluded from this within-patient analysis. a, Quantification of DEGs uniquely discovered using within-patient DE. The venn diagram represents the overlap of DEGs found using limma-voom and our within-patient DE approach. The bar plot describes the number of DEGs found per-patient using within-patient DE (right side) and the number of abnormal and normal cells per patient (left side). b, Volcano plot of 1,760 DEGs uniquely discovered using our within-patient DE approach. The y-axis represents the maximum -log10(q-value) of the gene across patients included in the within-patient analysis, and x-axis represents the maximum log2(fold change). The color and size of a dot denote the number of patients for which that DEG was detected, with blue dots representing DEGs detected in just one sample.

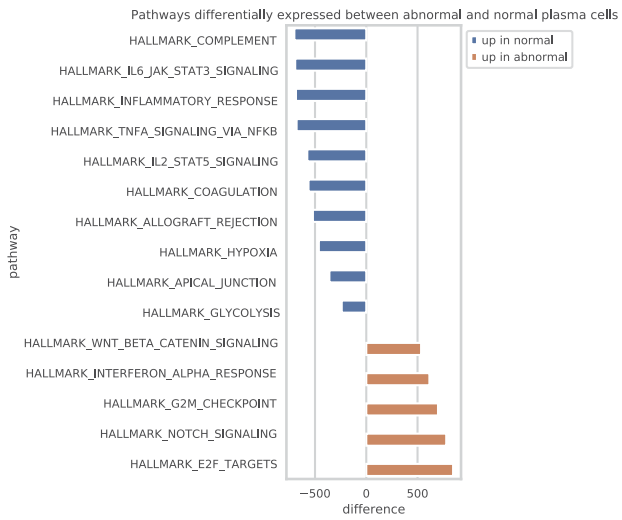


Figure 2.10: MSigDB hallmark genesets differentially enriched in abnormal samples (N=23 independent samples from 22 individuals) compared to normal (N=23 independent samples) (two-sided t-test, q<0.1). The difference between the mean enrichment among abnormal vs. normal samples is plotted on the x-axis.

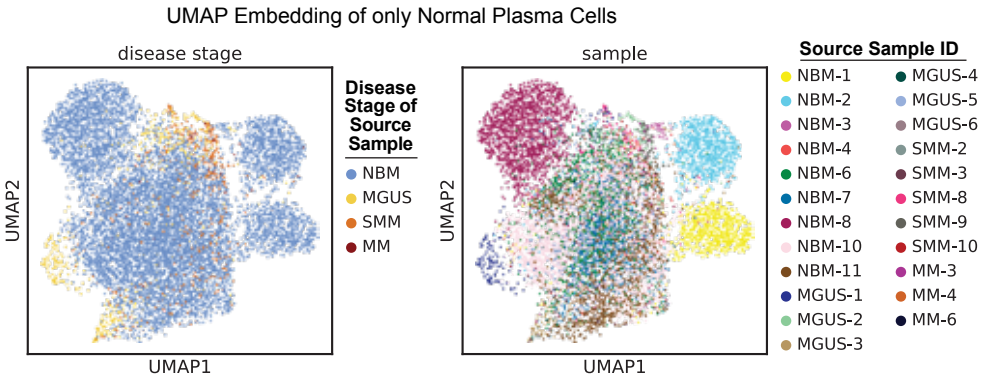


Figure 2.11: Inter-patient variation exists even among normal plasma cells (even among those from NBM donors), as observed in these UMAP embeddings containing only cells labeled as normal. The normal plasma cells on these plots are colored by the disease stage (left) and sample ID (right) of the patient from which they were extracted.

unique transcriptional profiles of individual tumors, which may not be shared across patients, without introducing the confounding effects that would arise from comparing tumor cells to normal cells from other healthy donors.

Of our eleven patients with both abnormal and normal cell populations, ten had significant DEGs detected between these populations ($|\log(\text{fold change})| > \log(1.5)$; false discovery rate $q<0.1$). Overall, this method identified 1,760 DEGs (full DEG lists were published as Supplementary Data in [30])), 1,509 of which were not found in our pseudobulked abnormal vs. normal DE analysis described above (Figure 2.9a). We found DEGs that are unique to individual patients (1,323 genes) as well as genes recurrently affected across patients, such as *CD27* (upregulated in 8 patients), *CD79A* (upregulated in 7 patients), and *RPL25* (downregulated in 7 patients). Many DEGs are found in samples across multiple disease stages (Figure 2.12).

We next focus on genes that were not discovered in the general DE analysis described earlier (Figure 2.9b). For example, within-patient DE enabled identification of significant upregulation of *FGFR3* and *WHSC1* in our patient with t(4;14). The general abnormal vs. normal cell comparison was not powered to identify this upregulation, since the translocation only occurred in a single patient in our cohort. Additionally, we discovered upregulation of *GNB2L1* (also known as *RACK1*; up in SMM-2, SMM-3, MM-6), a known oncogene in other cancers [48, 49] that has recently been reported to be upregulated in myeloma cell lines, [48, 49] but not yet in clinical samples. Among upregulated genes, we also found the histone gene *HIST1H1C* (MGUS-3, SMM-2, SMM-3, SMM-8, MM-6), the cell surface markers CD48 (MGUS-3, SMM-8) and *CD59* (MGUS-3, SMM-2, SMM-3, SMM-8, SMM-10), and the proto-oncogene *MYC* (MGUS-3, SMM-2, MM-6) (Figure 2.9b). We observed downregulation of *SSR4* (SMM-2, SMM-3, MM-3), associated with translocation of proteins across the endoplasmic reticulum, and *TPT1* (MGUS-3, SMM-8), a regulator of cell growth and proliferation. *ITM2C*, which has been reported for its expression on MM cells, [50, 51] was upregulated in some samples (MGUS-3, SMM-2, SMM-3, SMM-8) but downregulated in others (MGUS-6, SMM-9). While higher expression of *ITM2C* has been reported in patients with t(4;14) vs. without [52], we cannot conclude this from our data, as *ITM2C* was variably expressed in our 3 samples with t(4;14) (SMM-7, SMM-8, MM-1; Figure 2.13). Ribosomal proteins such as *RPS28* (SMM-2, SMM-3, SMM-10, MM-3, MM-6), RPLP1 (MGUS-3, SMM-2, SMM-3, SMM-10, MM-3, MM-4), *RPL14* (MGUS-3, SMM-2, SMM-3, SMM-10, MM-6), and others were recurrently upregulated, specifically in patients with hyperdiploidy. Although these ribosomal protein genes are upregulated in multiple samples, other samples have expression levels similar to those of NBM samples, possibly explaining why they were only detected using within-patient DE.

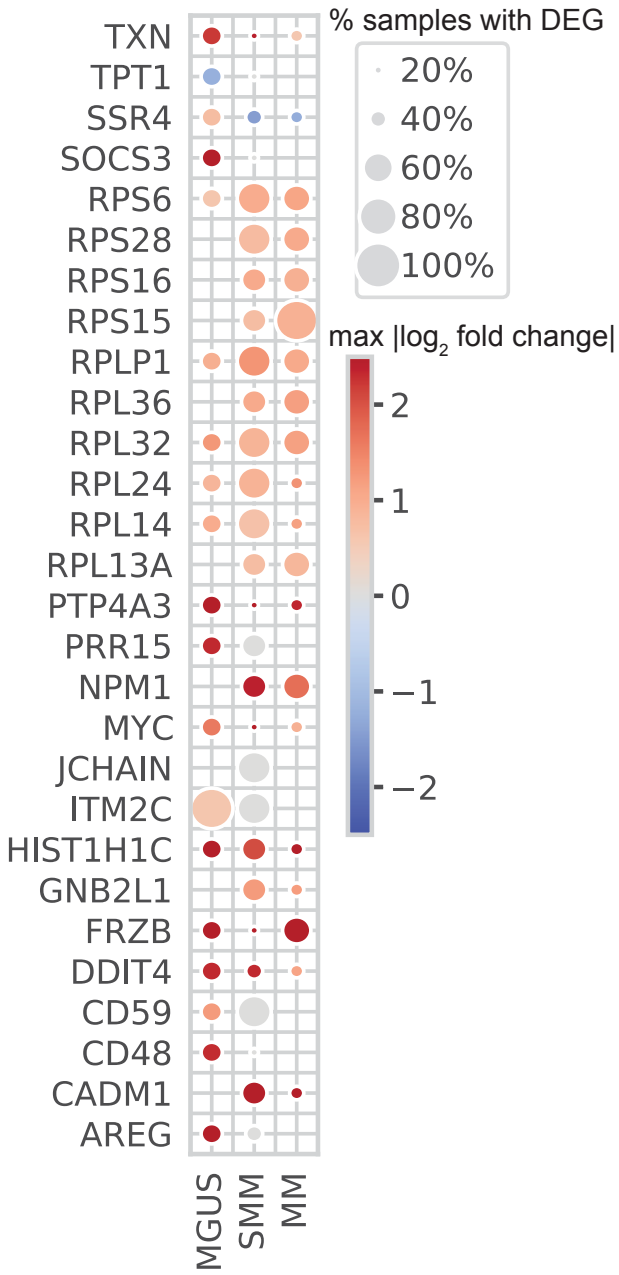


Figure 2.12: This heatmap portrays how differentially expressed genes (DEGs) were shared or varied across disease stages. The genes shown are the same genes annotated on the volcano plot in Figure 2.9b; these genes were uniquely discovered to be differentially expressed using our within-patient DE analysis, and not limma-voom. The size of the circle represents the fraction of samples from each disease stage in which that gene was determined to be differentially expressed ($|\log(\text{fold change})| > \log(1.5)$; $q<0.1$), out of a total of two samples that had DEGs from MGUS, five from SMM, and three from MM. The color of the dot represents the direction and magnitude of the greatest significant |log2 fold change| in each disease stage. For the purposes of the visualization, we cap the color bar at 2.5 and -2.5.

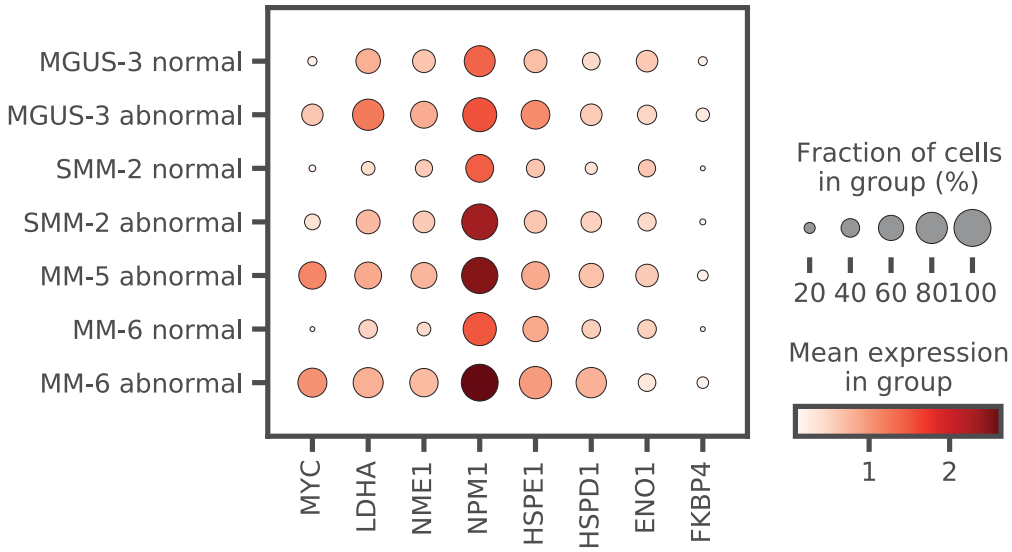


Figure 2.13: Expression of *MYC* and the *MYC* activation signature genes reported in Chng et al., 2011 in samples with significant DE of *MYC* by within-patient DE (MGUS-3, SMM-2, MM-6), as well as in MM-5, which was not included in the within-patient DE due to 100% purity, but had high levels of *MYC* comparable to MM-6. Cells are grouped by normal/abnormal labeling as well as sample ID (y-axis). Color intensity corresponds to the mean expression of the gene in each group, and dot size corresponds to the fraction of cells in the group that express the gene.

2.2.5 NMF discovers gene signatures that capture transcriptional programs

While our within-patient DE analysis allowed us to discover gene signatures in individual tumors, we next employed a method to discover gene signatures active in individual cells across our cohort, even if only in a small subset of cells in a tumor, and to characterize signature activity at the single cell level across disease stages.

Using our ARD-NMF method [6, 53], we decomposed the gene expression profiles across all plasma cells in our cohort into 28 gene signatures, each of which represents a pattern of gene expression recurrently occurring across cells in our dataset (Table A.4). Because we were most interested in highlighting signatures associated with disease biology rather than patient-specific effects, we removed signatures that were only active in a single patient. Similarly, since our goal was to find groups of genes with shared activity patterns, we did not focus our downstream analyses on signatures that only represented the expression of a single gene. After removing these "patient-specific" and "single-gene" signatures, we retained 15 gene signatures and examined the top genes from each signature to identify its underlying biological mechanism (Figure 2.14; Table 2.1). We tested to make sure that

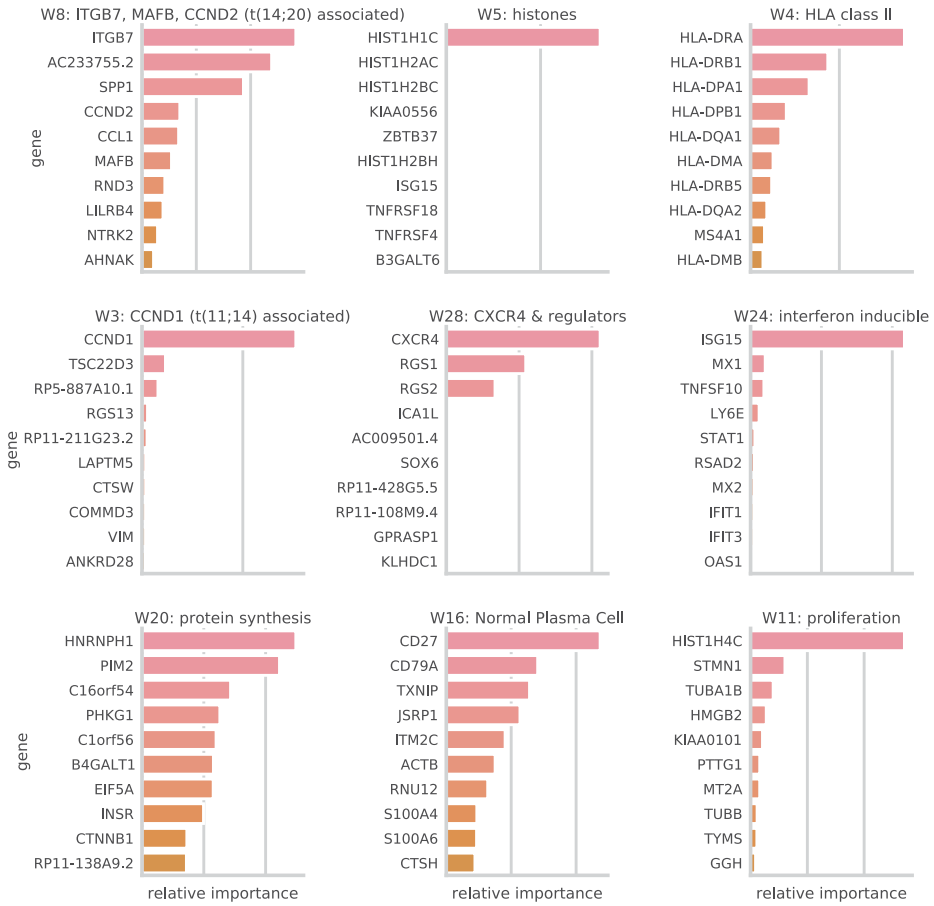


Figure 2.14: Bayesian non-negative matrix factorization uncovers gene signatures which capture myeloma cell biology across disease stages. Here, we show top genes for nine representative gene signatures. The importance score, plotted on the x-axis, is based on both the strength of the gene's contribution to the signature and its specificity to the signature.

signature activities did not correlate with batch variables, and found that they did not (details included in Section 2.4.12).

A number of the NMF signatures represent subtypes of myeloma that have previously been reported; for example, we find a signature of proliferation similar to that reported by Zhan et al. [54] and Broyl et al. [55], a *CCND1*-related signature which is differentially active in samples with t(11;14) (Figure 2.15), and a signature composed of *MAFB*, *CCND2*, and *ITGB7*, which is active in the samples with t(14;20).

We additionally discovered signatures that elucidate less well-characterized disease biology that is common to multiple samples in our cohort. For example, we found a signature with top genes *CXCR4*, which plays a role in normal plasma cell development and has also been implicated in MM progression [56, 57], and *RGS1* and *RGS2*, which are regulators of G Protein signaling that may regulate the CXCR4-CXCL12 axis [58]. We additionally found

Table 2.1: 15 gene expression signatures discovered using Bayesian NMF "patient-specific" and "single-gene" signatures are not included; see Table A.4 for the full list of signatures).

Signature	Biological Description	Top Genes
W3	t(11;14) associated	CCND1, TSC22D3, RP5-887A10.1, RGS13
W4	HLA class II	HLA-DRA, HLA-DRB1, HLA-DPA1, HLA-DPB1
W5	histones	HIST1H1C, HIST1H2AC, HIST1H2BC, KIAA0556
W8	t(14;20) associated	ITGB7, AC233755.2, SPP1, CCND2
W9	extracellular signaling	LGALS1, VIM, ACTB, S100A6
W11	proliferation	HIST1H4C, STMN1, TUBA1B, HMGB2
W16	normal plasma cell	CD27, CD79A, TXNIP, JSRP1
W20	protein synthesis	HNRNPH1, PIM2, C16orf54, PHKG1
W24	interferon inducible	ISG15, MX1, TNFSF10, LY6E
W28	CXCR4 & regulators	CXCR4, RGS1, RGS2, ICA1L
W1	unknown	JUNB, ZFP36, NFKBIA, IER2
W6	unknown	DUSP4, GADD45A, BTG2, LAMP5
W14	unknown	KLF6, TSC22D3, ANKRD28, KLF2
W26	unknown	HLA-A, ITM2C, PRR15, ACTB
W27	unknown	NEAT1, DDX17, ANKRD12, FOXO3

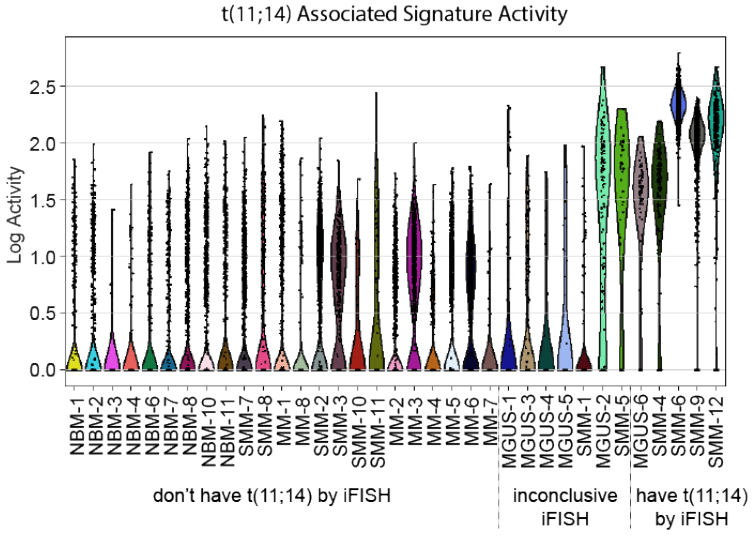


Figure 2.15: A signature with top contribution from *CCND1* is discovered and is most active in samples with t(11;14), as expected.

signatures that represent the activity of histone genes, interferon (IFN)-inducible genes, and genes involved in protein synthesis, among others.

2.2.6 Gene signature activity correlates with disease stage and microenvironment

For each gene signature, we tested whether its activity level varied between abnormal and normal cell populations or with disease stage, and discovered that three signatures had significantly different activity levels (Kruskal-Wallis $q < 0.1$ and Dunn's $q < 0.1$). As expected, we found that the t(11;14)-related signature is differentially active in the abnormal cell population from samples with the corresponding translocation compared to NBM cells. We found two additional signatures whose activity was significantly correlated with disease, described in detail below.

Abnormal cells across disease stages share universal downregulation of gene signature seen in normal PCs.

We discovered a "normal plasma cell signature" that is downregulated in abnormal cells at all stages of disease ($q = 3.3\times10\text{-}5$ and $1.0\times10\text{-}6$ for SMM and MM vs. NBM, respectively; while the same trend is observed between MGUS abnormal cells and normal plasma cells, it did not reach significance, likely due to low number of MGUS cases with sufficient abnormal cells). This signature robustly characterizes the normal bone marrow plasma cells in our data set (Figure 2.16a,b), and highlights genes that are downregulated only in the abnormal cell portions of samples at all disease stages. The top genes in this signature include *CD27* and *CD79A*, which are associated with the B cell lineage, and *JSRP1*, *CTSH*, *HCST*, and *RNU12*, genes as of yet unreported to be involved in plasma and MM cell biology (Figure 2.14; Figure 2.17a). Other canonical B cell markers were not expressed (*CD20*, *BCL6*, *PAX5*, and *E2F1*) on healthy PCs suggesting this effect is not due to B cell contamination (Figure 2.18a,b). Given the low tumor purity during early precursor conditions, this phenotype would be obscured at early disease stages in bulk samples; analysis at a single cell resolution, however, reveals that this healthy plasma phenotype is significantly downregulated in malignant cells as early as the MGUS stage (Figure 2.16a). Indeed, for our patient with serial samples at the SMM and MM stages (SMM-1 and MM-8), we find similarly low levels of signature activity at these two timepoints, underscoring the fact that this phenotype is lost at early stages of malignancy and remains low as the disease progresses (Figure 2.16a). Interestingly, the activity of this signature in normal cells also trends downward with increasing disease stage (Jonckheere-Terpstra test $p=1.3\times10\text{-}5$).

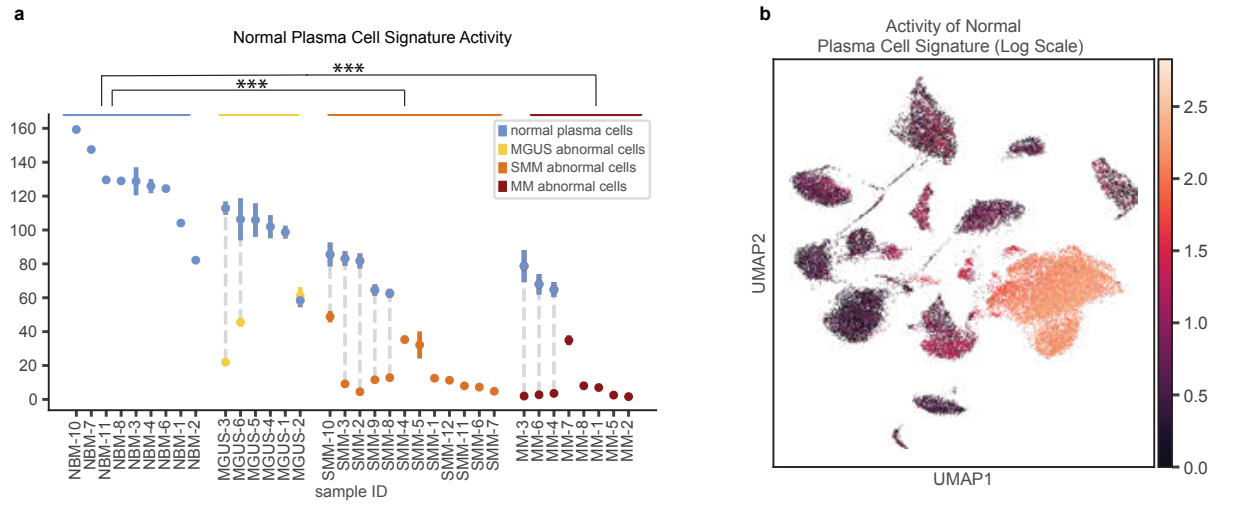


Figure 2.16: We discover a 'normal plasma cell signature' that is active in normal plasma cells across disease stages and downregulated in abnormal cells from MM and precursor conditions. We visualize this signature's activity by showing its mean activity ± s.e.m. for the normal and abnormal populations within each sample (a) and on a UMAP plot (log scale) (b). Mean activities were compared between groups, with *** denoting groups which significantly differ (abnormal cells from SMM (N=12) and MM (N=8), respectively, vs. NBM (N=9)).

The individual top genes on this signature are also downregulated in patients' abnormal cells compared to healthy cells (Figure 2.17b). The notable exception is our CD20+ sample, SMM-12: while CD27 is upregulated in abnormal cells vs. normal cells in this sample, the NMF normal plasma cell signature nonetheless has low overall activity (Figure 2.16a), demonstrating the universal loss of this signature across tumors with different phenotypes.

Validation of normal plasma cell signature in independent datasets.

To validate our findings, we ran the ARD-NMF algorithm on single cell data from Ledergor et al. [28] and recovered a similar signature with top genes CD27, CD79A, and JSRP1. This signature, too, is strongly downregulated in abnormal cells at all disease stages ($q = 2.4\times10\text{-}4$ and $1.5\times10\text{-}5$ for SMM and MM vs. healthy donors respectively; only one MGUS sample had abnormal cells, and it too appears to be downregulated; Figure 2.19). As additional validation in bulk data, we estimated the activity of our normal plasma cell signature in bulk RNA sequencing from newly diagnosed MM patients in the Multiple Myeloma Research Foundation's (MMRF) CoMMpass dataset for their expression of this signature, and also estimated tumor purity in these samples (methodological details included in Section 2.4.13). We found a significant negative correlation between signature expression and tumor purity, further supporting this signature as a marker of normal plasma cells (Figure 2.17c).

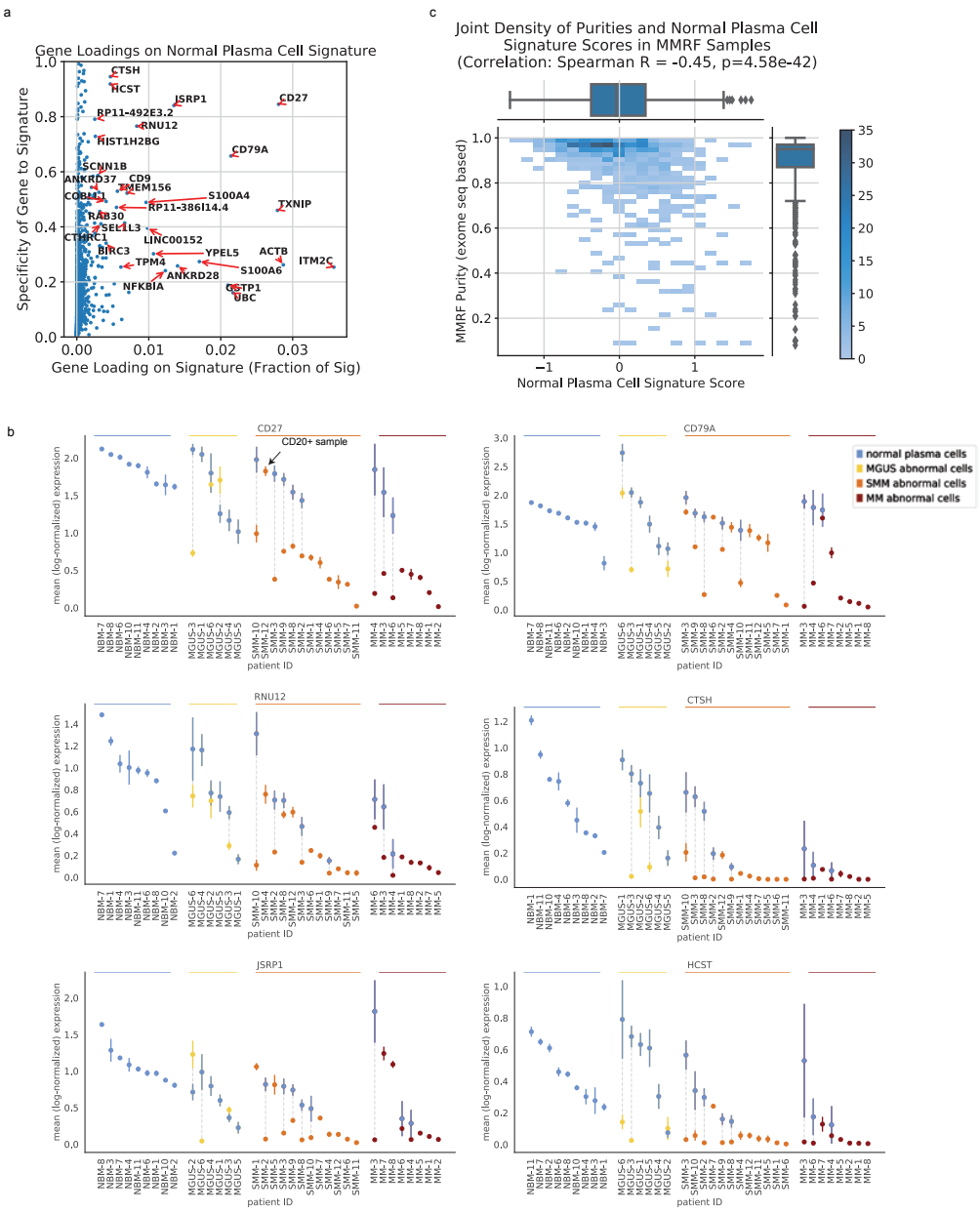


Figure 2.17: a, Contributions of individual genes to the 'normal plasma cell signature.' The x-axis represents a gene's loading on the signature, i.e. its value in the W matrix, and the y-axis represents a gene's specificity to the signature (see Section 2.4.11). b, Mean expression ± s.e.m. of top genes from the 'normal plasma cell signature' in abnormal and normal plasma cell portions of samples. Expression of the top genes from the signature are generally also downregulated in abnormal cells as compared to normal plasma cells at all stages of disease. For *CD27*, the one notable exception is sample SMM-12, which has a CD20+ phenotype. c, In bulk samples from the MMRF dataset (N=826 independent samples), we observe a significant negative correlation between sample purity and 'normal plasma cell signature' score (Spearman R=-0.45, p=4.58×10-42). The joint density of purity and signature score is plotted, with the intensity of color indicating the number of samples at a given part of the distribution (see color bar). Boxplots representing the marginal distributions of signature scores and purities are plotted along the top and right, respectively (center line, median; box limits, upper and lower quartiles; whiskers, 1.5x interquartile range; points, outliers).

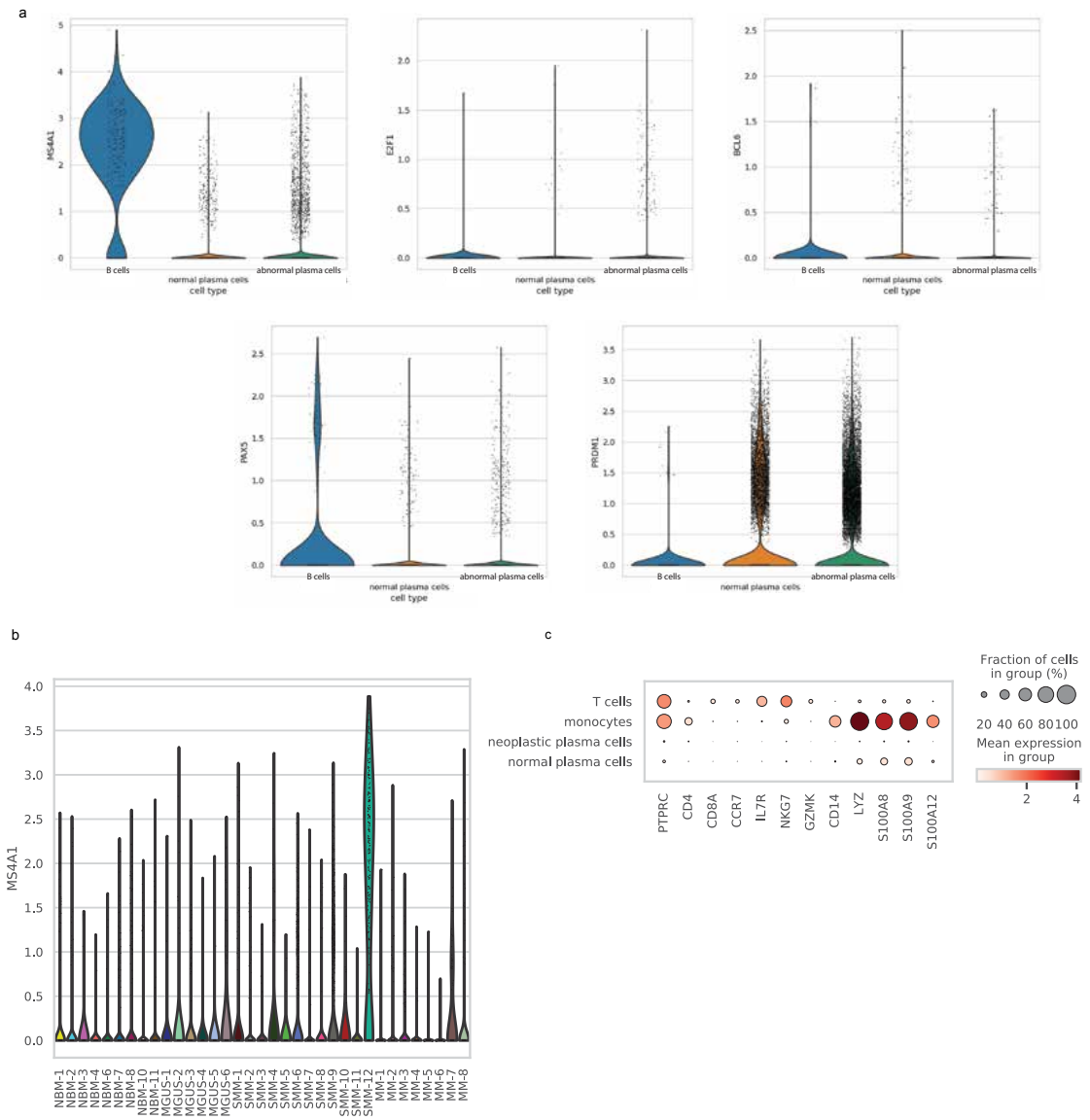


Figure 2.18: No contamination from B cells, T cells, or monocytes in CD138+ cells. a, B cell surface marker *CD20* (also known as *MS4A1*) and B cell transcription factors *E2F1*, *PAX5*, and *BCL6* are virtually not expressed in CD138+ cells (with the exception of *CD20* expression in SMM-12; see (b)), while they are expressed on the B cells which were removed from our data as part of QC. The plasma cell transcription factor *PRDM1* is expressed on CD138+ cells, but not B cells. Violin plots show distribution of expression over cells in each group. b, Expression of *CD20* on CD138+ cells is largely driven by patient SMM-12, who has a CD20+ MM phenotype. c, T cell and monocyte marker genes are hardly expressed in our CD138+ normal or abnormal cells, while they are expressed in the T cell and monocyte populations which we removed from our data as part of QC. While there are low levels of monocyte marker genes among normal PCs, indicating possible low levels of ambient contamination in some normal samples, these genes are not expressed in abnormal cells, indicating that abnormal PCs (i.e. the PCs plotted in Figure 2.20 for comparison with monocytic populations) are uncontaminated. Color intensity corresponds to the mean expression of the gene in each group, and dot size corresponds to the fraction of cells in the group that express the gene.

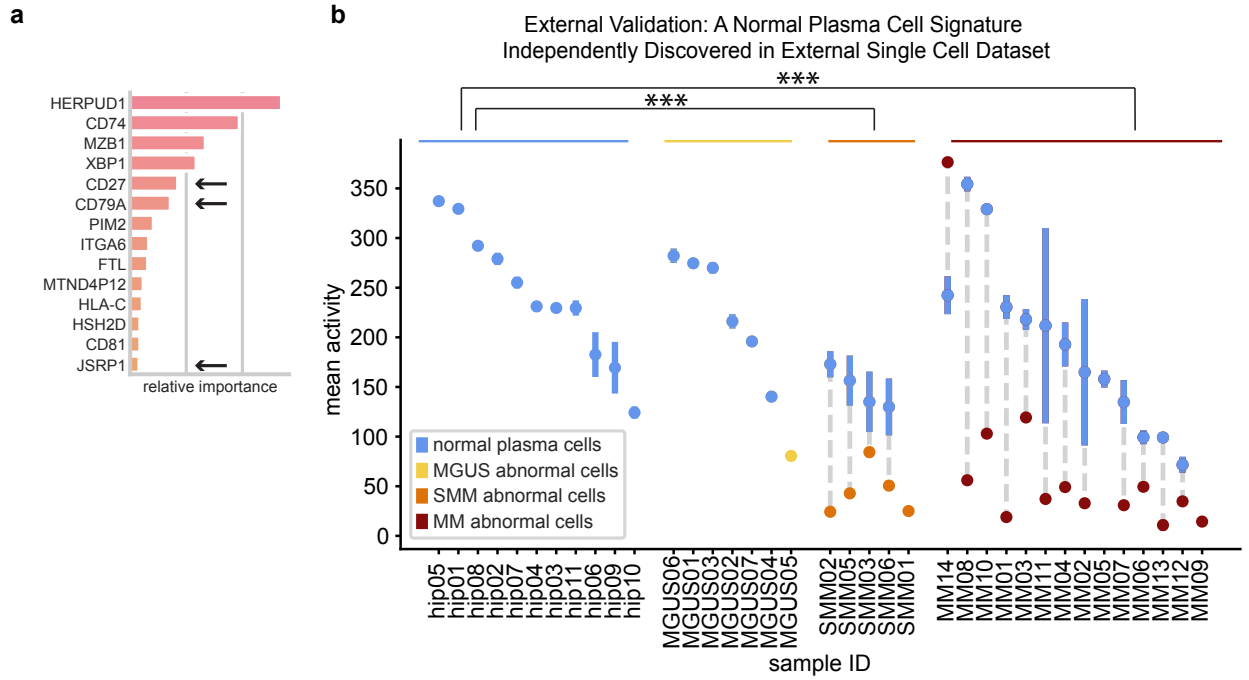


Figure 2.19: a, Validation on external dataset: our NMF algorithm run on external CD138+ single cell data from MGUS, SMM, MM and healthy donors independently discovers a gene signature similar to our normal plasma cell signature, with shared top genes CD27, CD79A, and JSRP1. b, After labeling cells in that dataset as normal or abnormal, we discover that this signature follows the same pattern as in our data, with high activity in normal cells and a significant decrease in activity in abnormal cells across disease stages. Mean activities ± s.e.m. across cells in normal and abnormal portions of samples are shown, with *** denoting groups which significantly differ (abnormal cells from SMM (N=5) and MM (N=13), respectively, vs. NBM (N=11)).

Interferon-inducible signature upregulated in tumor and microenvironment.

We discovered a signature enriched for IFN-inducible genes, such as *ISG15* ubiquitin like modifier (*ISG15*), MX Dynamin Like GTPase 1 (*MX1*), and Interferon Induced Protein With Tetratricopeptide Repeats 1 and 3 (*IFIT1* and *IFIT3*) [59] (Figure 2.14). Notably, this signature is significantly upregulated in both normal and malignant populations from overt MM patients compared to NBM ($q = 5.2\times10\text{-}3$ and $q = 3.2\times10\text{-}4$, respectively; Figure 2.20). This upregulation is specific to malignant disease (signature is not significantly upregulated in precursor conditions; one patient, SMM-11, is an outlier with very high activity). Further, a previous study [31] discovered similar IFN-inducible signatures when running ARD-NMF on T cell and CD14+ monocytes from the microenvironments of these patients' tumors, and these signatures are active in the same patients that have high IFN-inducible gene expression in their CD138+ cells (Figure 2.21). T cell and monocyte markers were not expressed in CD138+ cells, suggesting this correlation is not due to cell type contamination (Figure 2.18c).

We briefly note that two gene signatures discovered in the CD138+ cells involved interferon-inducible genes: W24 (the plasma cell "IFN inducible" signature), whose activity is shown in the topmost heatmap in Figure 2.21, and W7, a patient-specific signature which involves many varied genes highly expressed in sample MM-4, including IFI27 and IFI6. Because W7 also involves non-interferon related genes, we do not show it in the main figure, but MM-4's high expression of interferon-inducible genes is captured by that signature (mean activity of W7 in MM-4 = 276) instead of W24.

2.2.7 Tumors contain transcriptionally heterogeneous cell subpopulations

The NMF approach to signature discovery allows us to find groups of genes with shared activity in single cells, and thus not only to examine how signature activity varies between samples and disease stages, but also between subpopulations of abnormal cells within a single sample. Indeed, we find that tumors are heterogeneous, with subpopulations of cells expressing distinct subsets of the NMF gene signatures we discovered (for methodological details, see Section 2.4.16). For example, considering our samples from myeloma patients, in MM-1, disjoint subsets of cells express the IFN-inducible, proliferation, extracellular signaling and protein synthesis signatures; this is discernible on a UMAP plot of patient MM-1's cells colored by the activity level of these signatures or the top genes from these signatures (Figure 2.22; Figure 2.23, Figure 2.24). Similarly, MM-2 contains a subset of cells with high expression of the protein synthesis signature, MM-4 contains a subset of proliferating cells, and MM-5 contains a subset of proliferating cells as well as cells with varying activity of

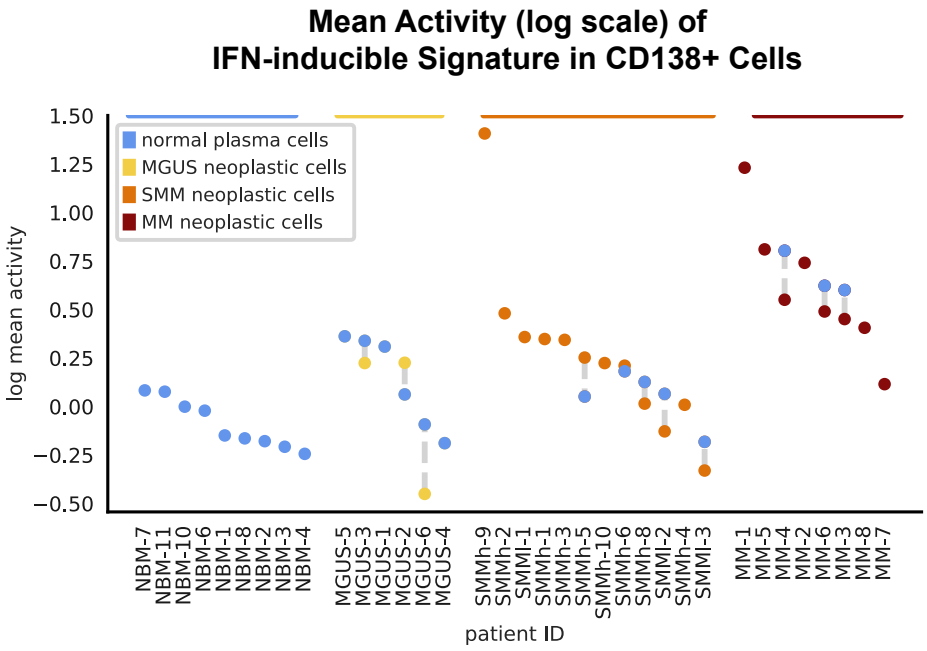


Figure 2.20: Mean activity ± s.e.m. of plasma cell IFN-inducible signature across normal and abnormal plasma cell populations. Both normal and abnormal plasma cells exhibit significantly increased activity of the interferon-inducible signature in MM vs. normal bone marrow (q=5.2×10-3 and q=3.2×10-4, resp.).

signature 22 (a single-gene signature representing the expression of TMSB4X) (Figure 2.22; Figure 2.23b,c,d, Figure 2.24). The NMF signatures that were found to be heterogeneous in multiple samples were those relating to HLA class II genes (heterogeneous in 2 samples), extracellular signaling genes (heterogeneous in 2 samples), proliferation genes (heterogeneous in 8 samples), protein synthesis genes (heterogeneous in 2 samples) and IFN-inducible genes (heterogeneous in 2 samples).

2.3 Discussion

Early identification of precursor MM conditions and those patients that are at risk of progressing to overt MM is critical as it could allow for early therapeutic interventions in these patients. However, the current risk criteria used to identify high risk precursor patients who would most benefit from treatment are mostly based on clinical parameters such as M spike, light chains or percent tumor burden [20]. Therefore, elucidation of the molecular transformation that occurs at early tumorigenesis and later at high risk SMM before disease progression is critical for developing informed criteria for patients who would benefit from early intervention and targets that may be exploited for therapeutics [22, 23, 61]. Here,

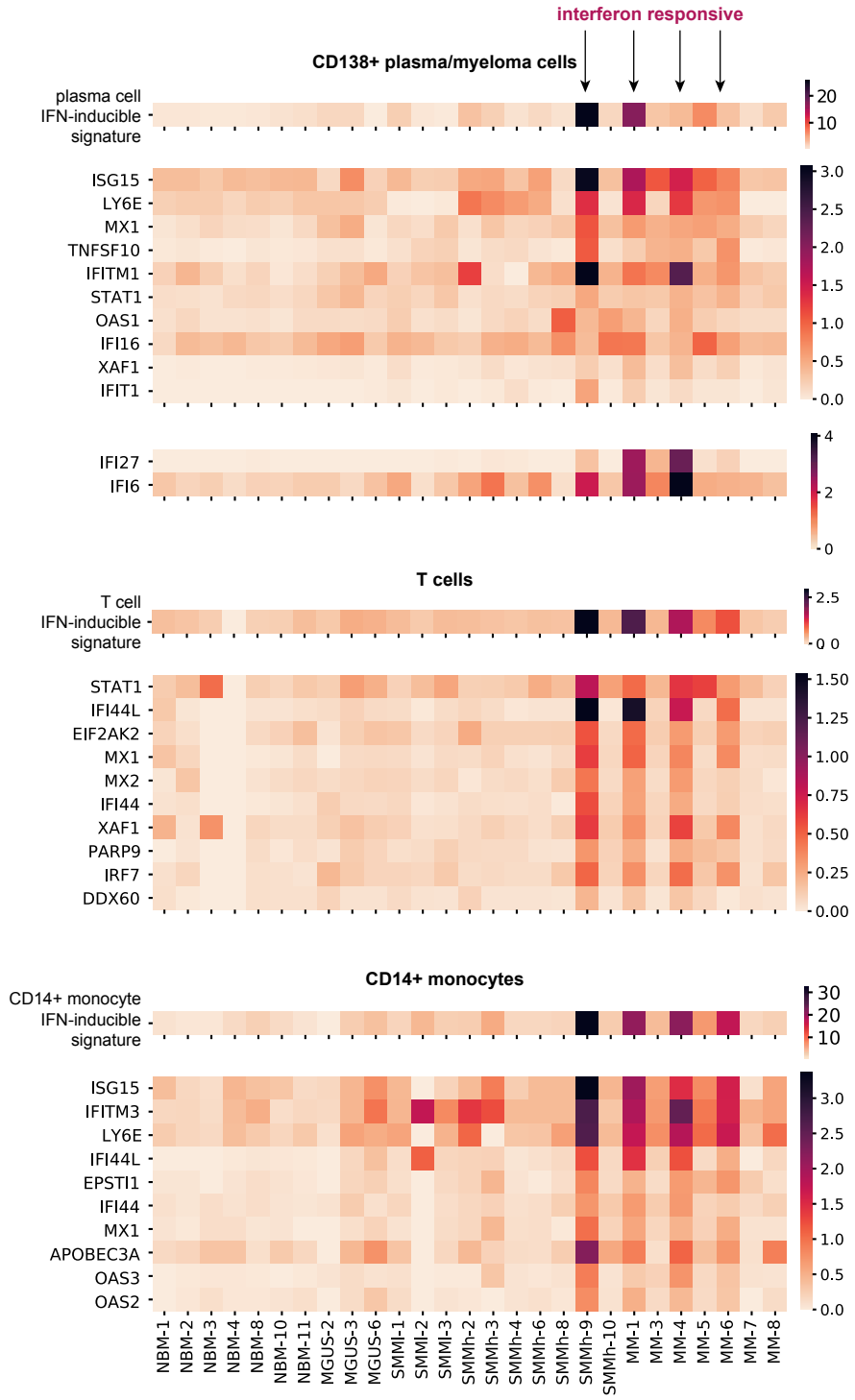


Figure 2.21: IFN-inducible signature is correlated between CD138+ and microenvironment cells. Mean activity per sample of IFN-inducible signature discovered in CD138+ cells (top), T cells (middle) and CD14+ monocytes (bottom). NMF signature results and expression data for T cells and monocytes were taken from Zavidij et al. [60]. Mean expression levels for the ten genes with the highest values in the W matrix for each signature are also shown. Expression of additional interferon-inducible genes *IFI27* and *IFI6* is shown for CD138+ samples.

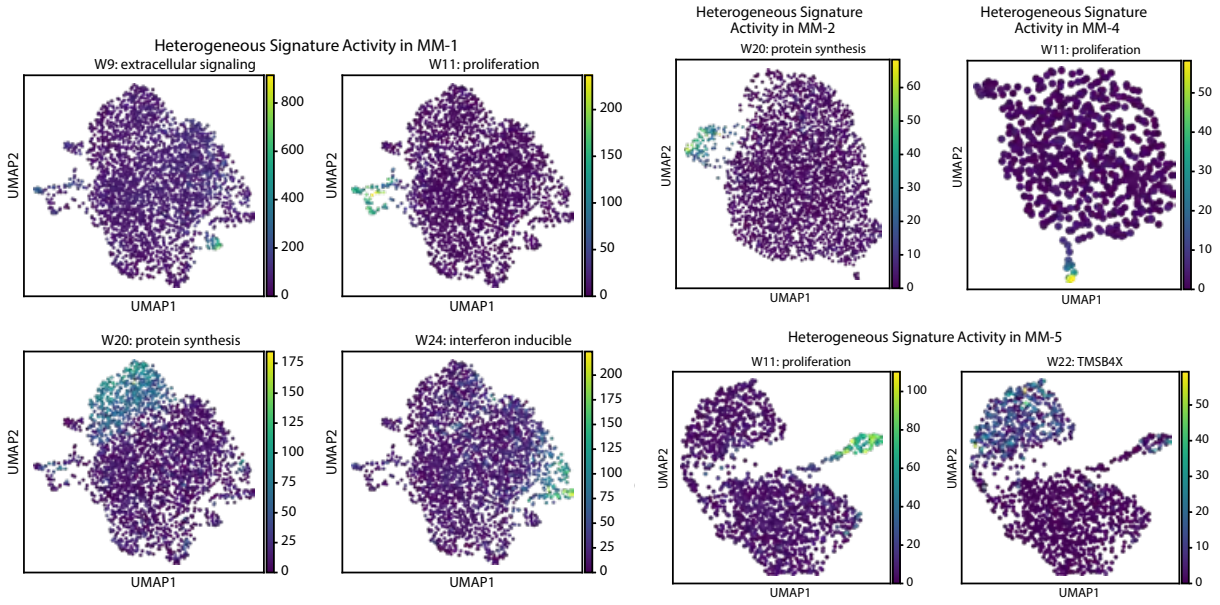


Figure 2.22: Subpopulations within patient tumors heterogeneously express gene signatures. Cells from a given MM sample were projected onto a UMAP plot based on expression of highly variable genes, and colored by the activity level of NMF signatures determined to be heterogeneously expressed in that sample.

we leveraged single-cell RNA sequencing to overcome the challenges of characterizing the transcriptomics of these low burden precursor states. While some patients at early stages of disease had low disease burden, such that their driving cytogenetic translocations could not yet be detected by iFISH in the clinic, we demonstrate that RNAseq is sensitive enough to already reveal their underlying cytogenetic changes. This result highlights the possibility of using RNAseq to detect phenotypic changes in patients' bone marrow plasma cells earlier than methods currently used in the clinic [62].

Precise labeling of normal and abnormal cells in each sample revealed low tumor purity in samples from earlier disease stages, even when subsetting to only CD138+ cells. This suggests that conclusions drawn from bulk studies of precursor conditions are likely influenced by heavy contamination by normal plasma cells. For example, Chng et al. [27] conclude that their "MYC activation signature" is upregulated in a subset of myelomas, but not in MGUS. While it is possible that *MYC* activation really did not occur in their MGUS samples, we do find *MYC* and some of their *MYC* activation signature genes to be significantly upregulated in two precursor patients in our cohort (MGUS-3 and SMM-2), as well as in two MM patients (MM-5 and MM-6). The upregulation of *MYC* in MGUS is clinically relevant, as it is associated with tumor aggression, poor clinical outcomes, and potentially with disease progression [25, 26, 63]. Both the low tumor purity in MGUS and the potential rareness of

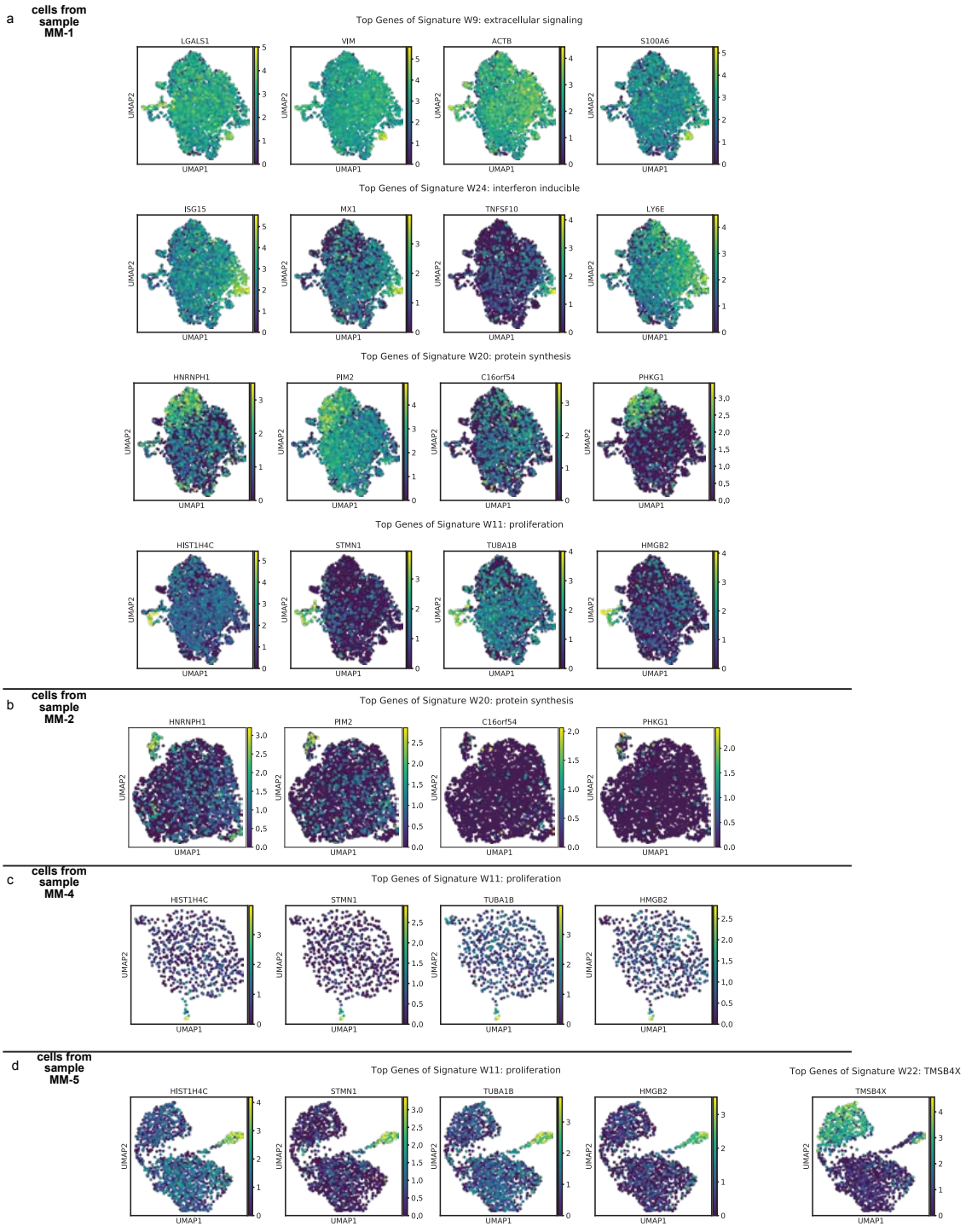


Figure 2.23: Genes are heterogeneously expressed within tumor samples. UMAP embeddings of cells from samples MM-1 (a), MM-2 (b), MM-4 (c) and MM-5 (d), colored by expression of the top 4 genes from each signature that was highlighted as heterogeneously active within that patient in Figure 2.22. Since signature 22 was a single-gene signature, we only plotted the expression of the top gene (*TMSB4X*).

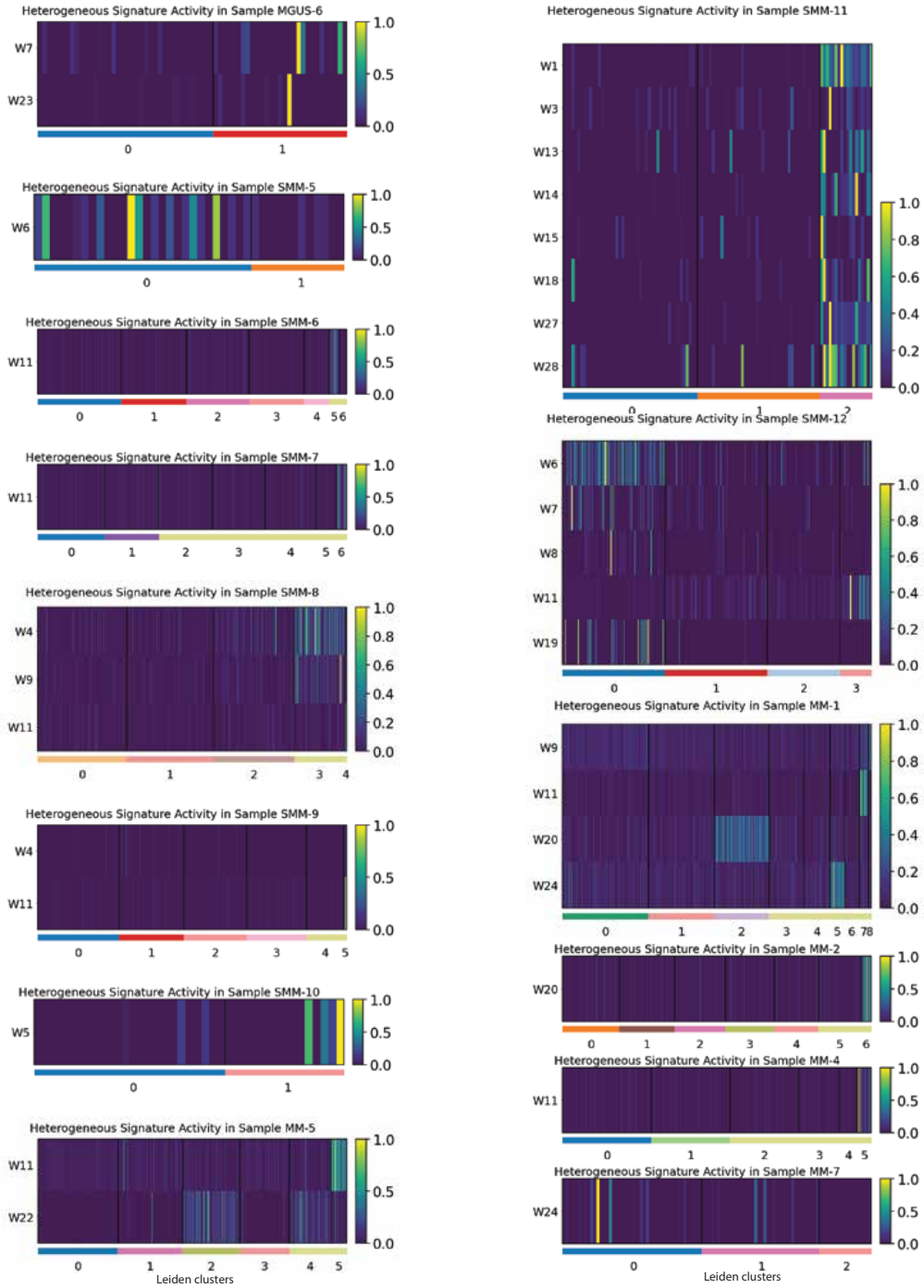


Figure 2.24: Heatmap of NMF signature activity x cells in each sample, for signatures that were found to be heterogeneously active within that sample. Rows are standardized (for each row, we subtracted the minimum and divided by its maximum). Cells are ordered according to their cluster assignment (these are the clusters which were used to determine heterogeneous signature activity, as described in Section 2.4.16; they are distinct from other clusterings mentioned throughout this study).

this phenotype among MGUS patients would have made this difficult to discover without single cell data, the ability to distinguish normal vs. abnormal cells, and our within-patient DE analysis.

Through our isolation of abnormal cells, we found that these early precursor conditions already exhibit transcriptomic alterations seen in overt MM. As one important example, we identified a signature present in normal plasma cells but uniformly lost at all stages of MM progression. *CD27*, one of the top genes of this signature, has been previously discussed in MM literature, but has been reported to have variable expression in myeloma cells, increased expression in MGUS, and a correlation with prognosis [37–39]. Our data shows significant downregulation of *CD27* compared to normal plasma cells as early as the MGUS stage (Figure 2.13). Although we observe a trend of decreased *CD27* expression in MM compared to SMM (Figure 2.13), it raises the question of the extent to which previous results were confounded by increasing tumor purity as the disease progresses. This would need to be tested in a larger cohort with single cell data.

In addition to *CD27*, abnormal plasma cells also had lower expression of another mature B cell marker, *CD79A*, as well as decreased enrichment of immune pathways, such as complement pathway (including decreased expression of the complement receptor 2, *CR2*, also known as *CD21*). Similarly, other studies found absence or low levels of B cell surface markers *CD19*, *CD27*, and *CD45* on abnormal cells compared to normal plasma cells [37, 64]. Our study extends the characterization of matched abnormal and normal plasma cells to the whole transcriptome. It also supports the hypothesis that the loss of B cell immune functionality, as assessed by gene expression programs and cell surface protein expression, is an early step in the generation of tumor plasma cells.

When probing pathway level transcriptional changes in abnormal cells, we found aberrant expression of *Wnt* pathway members including overexpression of *DKK1* in abnormal cells of precursor myeloma. *DKK1* is secreted by myeloma cells and is associated with the presence of osteolytic lesions through inhibition of osteoblast differentiation [65, 66]. Given that many cases of MGUS also have osteoporosis and osteopenia, this may indicate that *Wnt* dysregulation and *DKK1* overexpression are associated with early osteopenia in those patients and are potentially predictors of the development of osteolytic lesions.

When investigating connections between the phenotype of PCs and dynamics in the tumor immune microenvironment, we discovered that patients who exhibit upregulation of an *IFN*-inducible signature in their tumor cells exhibit this same phenotype in their normal bone marrow PCs, as well as in T cells and monocytes in their tumor microenvironment. This suggests that interferon signaling in myeloma cells, which has been reported previously [67], may be a response to a common stimulus in the microenvironment that affects multiple

cell types, including normal plasma cells. Further work is needed to investigate the potential common mechanisms driving the upregulation of interferon signaling across these cell types.

The single cell resolution of our data provides insight into the heterogeneity of signatures within patient samples. For example, while the proliferation signature has previously been reported in bulk studies as characterizing a distinct subset of patients [54], our data reveal that in fact, only a subset of cells from any given patient exhibit a proliferative signature. Additionally, these proliferating cells may be found in patients that harbor the driver mutations previously used to characterize patients that belong to the non-proliferative subtypes.

Finally, our within-patient DE analysis points to potential therapeutic targets that are present in subsets of patients not only in MM, but also in earlier disease stages, which can be selected for functional validation. We found that select patients exhibit upregulation of genes associated with the proteasome, which may correlate with sensitivity to bortezomib and antitumor immune response [68]. While certain proteasome genes were identified in our general abnormal vs. normal DE analysis (*PSMB4*, *HSPB1*), we discovered upregulation of additional proteasome genes in select patients using our within-patient DE approach (e.g. *PSMA4*, *PSMD14*, *PSMD11*, *PSMC6*, and *PSMA1* in MGUS-3 and SMM-8), painting a much fuller picture of proteasome-related gene enrichment in those samples. Additionally, this approach allowed us to detect that *CD59*, a complement inhibitor whose expression has been associated with resistance to daratumumab in myeloma [69] and to anti-CD20 therapies in B cell malignancies [70], was significantly upregulated in five patients, including those with precursor conditions. In a similar vein, we discovered upregulation of *CD48*, which has been nominated as a drug target in MM [71], in abnormal cells from MGUS and SMM. Our identification of patient-specific transcriptional changes as early as MGUS paves the way for future work exploring personalized treatment approaches prior to malignant disease.

In summary, our work used single-cell RNA sequencing to overcome the low fraction of abnormal plasma cells in early precursor disease and uncovered previously unidentified transcriptional changes that occurred at the precursor stage and could not be described in prior bulk sequencing studies. We identified previously-unappreciated commonalities between MGUS and overt MM, such as the loss of our novel normal plasma cell signature. Elucidating patient-specific transcriptional changes, as our within-patient DE analysis does, may enable precision medicine approaches for treating MM and potentially intercepting precursor conditions before progression.

2.4 Methodological Details

2.4.1 Patient samples and cell preparation

BM samples from patients with MGUS, SMM or MM, as well as healthy donors were collected as approved by the Dana-Farber Cancer Institute Institutional Review Board. Informed consent was obtained from all patients and healthy volunteers in accordance with the Declaration of Helsinki protocol (fifth revision from 2000 with Clarifications of Articles 29, 30 (20022004), and the most recent iteration from 2013). MGUS and SMM patient samples were collected for a clinical trial, clinicaltrial.gov identifier NCT02269592. CD138+ BM cell fractions were isolated using magnetic-activated cell sorting technology (Miltenyi Biotec). Selected cells were either viably cryopreserved in dimethylsulfoxide at a final concentration of 10% or used immediately for scRNA-seq.

2.4.2 Sequencing library construction using the 10x Genomics platform

Frozen BM cells were rapidly thawed, washed, counted and resuspended in PBS and 0.04% bovine serum albumin to a final concentration of 1,000 cells per µl. The Chromium Controller (10x Genomics) was used for parallel sample partitioning and molecular barcoding. To generate a single-cell Gel Bead in EMulsion, cellular suspensions were loaded on a Single Cell 3' chip together with the Single Cell 3' Gel Beads, according to the manufacturer's instructions (10x Genomics). scRNA-seq libraries were prepared using the Chromium Single Cell 3' Library Kit v.2 (10x Genomics). Fourteen cycles were used for the total complementary DNA amplification reaction and for the total sample index PCR. Generated libraries were combined according to Illumina specifications and paired-end sequenced on HiSeq 2500/4000 platforms with standard Illumina sequencing primers for both sequencing and index reads; 100 cycles were used to sequence Read1 and Read2.

2.4.3 Preprocessing of scRNA-seq data

Sample demultiplexing, barcode processing, alignment to the human genome (hg38) and single-cell 3' gene counting was performed using the Cell Ranger Single-Cell Software Suite v.2.0.1. Cells called by Cell Ranger were further filtered to those with $<$15% mitochondrial expression, $>$200 genes covered, $<$50,000 total unique molecular identifiers (UMIs), and

<4,000 total genes detected. Log-normalized expression values were calculated as:

$$e_{g,c} = \log\left(\frac{10^4}{N_c} n_{g,c} + 1\right)$$

for a cell c with Nc total UMIs from genes (excluding genes that accounted for >20% of UMIs in any cell), with $n_{g,c}$ UMIs mapping to gene g. Except where noted, "expression" refers to log-normalized expression.

2.4.4 Gene selection prior to downstream analysis

For downstream analyses (PCA, UMAP, Leiden clustering, differential expression, NMF), we removed genes located in *IGH*, *IGL*, or *IGK* loci (based on the GRCh38 reference), since these are expected to be upregulated and clonally expressed in abnormal cells, dominating other transcriptional disease signatures of interest. Sex genes *XIST* and *RPS4Y1* (the two genes with the greatest absolute fold changes when comparing gene expression in male vs. female samples in our cohort) were also removed prior to PCA, UMAP, clustering, and NMF, so as not to separate samples based on the sex of the patient but rather based on disease biology. Highly variable genes were selected based on log-normalized expression data using the `highly_variable_genes` function in Scanpy (version 1.7.1) [72] with default parameters and `max_mean=4`, except where indicated otherwise.

2.4.5 Removing non-CD138+ cell populations

To remove cells incorrectly sorted during bead selection, we first performed coarse clustering of all cells sorted as CD138+. We centered and scaled the data, clipping the resulting values to a maximum of 10, calculated highly variable genes, projected the expression of highly variable genes onto its first 14 principal components, and computed Leiden clusters (resolution=1.5), all using the Scanpy (version 1.7.1) [72] package with default parameters except as specified. We chose this high resolution for clustering as our goal was to find and remove even small clusters of contaminating non-CD138+ cells. Using expression of known cell type markers, we identified and removed clusters of cells containing non-CD138+ immune cells, red blood cells, and cells from the extracellular matrix.

2.4.6 Leiden clustering of CD138+ cells

After removing contaminating cell types, we reprocessed our data prior to downstream analyses. Despite removing clusters of red blood cells, we still detected ambient contamination of

hemoglobin genes in some samples, and thus we regressed out a "hemoglobin score," computed as the mean of log-normalized expression of hemoglobin genes. We then recomputed highly variable genes, re-centered and scaled the data, clipping the resulting values to a maximum of 10, projected the expression of highly variable genes onto its first 14 principal components, and computed Leiden clusters (resolution=1.5), all using the Scanpy package (version 1.7.1) [72] with default parameters except as specified. Here, we again chose a high resolution for clustering in order to detect even small clusters of unique cell types. We merged seven clusters which were all determined to contain healthy cells based on the majority of cells in these clusters coming from NBM samples, their overexpression of genes such as *CD27*, and their co-localization on a 2D UMAP embedding plot.

2.4.7 Bayesian Model for Sample Purity Estimation

We developed the following hierarchical Bayesian model to automatically estimate ρ, the purity of a sample, based on the number of cells expressing the kappa immunoglobulin gene and the total number of cells in the sample. This model is based on the rationale that since abnormal cells are descended from a single B-cell progenitor with a specific V(D)J rearrangement they will contribute either all kappa or all lambda-expressing cells. In contrast, the non-abnormal cells are a highly diverse group of cells with a ratio of kappa versus lambda immunoglobulins reflective of the relative frequencies of these rearrangements. By analyzing our NBM samples, we observe that the fraction of non-abnormal cells with kappa rearrangements are similar across individuals, albeit with some variance which we model with a truncated normal distribution (between 0 and 1). The measured frequency of cells with kappa vs. lambda rearrangements is a function of the mixture of cells originating from these two distributions. We can therefore use the observed counts in each sample to calculate the probability that a sample is composed of a given proportion of normal and abnormal cells (i.e. estimate sample purity). Specifically, we assume the following generative model,

$$
\begin{aligned}
\kappa_n &\sim \text{Truncated Normal}(\mu, \sigma^2, 0, 1) \\
\kappa_t &\sim \text{Bernoulli}(0.5) \\
\rho &\sim \text{Beta}(1, 1) \\
p &= \rho * \kappa_t + (1 - \rho) * \kappa_n \\
n_\kappa &\sim \text{Binomial}(N, p)
\end{aligned}
$$

where κ_n is the proportion of kappa cells among normal cells in a sample, κ_t is the

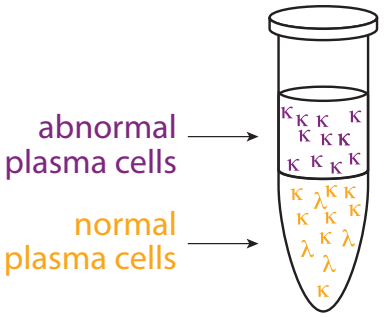


Figure 2.25: Samples are mixtures of an abnormal cell population and a normal cell population. Each cell in a sample expressed either a kappa or lambda immunoglobulin light chain.

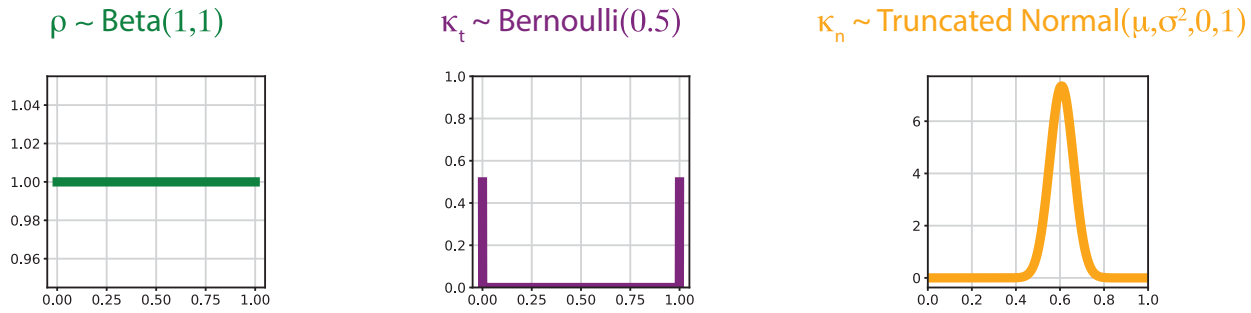


Figure 2.26: The three prior distributions used in the generative process for our Bayesian purity model: (left) the ratio of normal to abnormal cells in a sample, i.e. the sample purity ρ, is drawn from a uniform prior; (center) for the abnormal cell population, the proportion of cells expressing *IgK* is either 0 or 1 (due to clonality), and thus is drawn from a Bernoulli distribution with 50% probability; (right) for the normal plasma cell population, the proportion of cells expressing *IgK* is drawn from a normal distribution truncated between 0 and 1. μ and σ^2 were determined empirically based on the normal bone marrow samples in our cohort.

proportion of kappa cells among abnormal cells in a sample (either 0 or 1 due to clonality), ρ is sample purity (drawn from a uniform distribution), p is the proportion of kappa cells in a full sample (i.e. after mixing normal and abnormal cells), N is the total number of cells in a sample, and n_κ is the total number of kappa cells in a sample. κ_n, κ_t, and ρ are unobserved, p is a deterministic function of these, and both N and n_κ are observed. Cells were defined as kappa or lambda based on whether they have higher expression of *IGKC* or *IGLC2*, respectively. μ and σ^2 are empirically estimated from the NBM samples in our cohort. Since we do not assume prior knowledge about ρ, we use an uninformative Beta(1,1) (i.e., uniform) prior. We assume that a patient is equally likely to have a kappa or lambda myeloma (thus the Bernoulli 0.5 distribution for κ_t).

We calculate the posterior probability of the sample purity, ρ, given the number of kappa

cells and the total number of cells in the sample:

$$P(\rho|n_\kappa, N) \propto P(\rho|N)P(n_\kappa|\rho, N) \tag{1}$$

$$= P(\rho|N) \sum_{\kappa_t \in \{0,1\}} \int_0^1 P(n_\kappa, \kappa_n, \kappa_t|\rho, N)\, d\kappa_n \tag{2}$$

$$= P(\rho|N) \sum_{\kappa_t \in \{0,1\}} \int_0^1 P(n_\kappa|\kappa_n, \kappa_t, \rho, N)P(\kappa_n, \kappa_t|\rho, N)\, d\kappa_n \tag{3}$$

$$= P(\rho) \sum_{\kappa_t \in \{0,1\}} \int_0^1 P(n_\kappa|\kappa_n, \kappa_t, \rho, N)P(\kappa_n)P(\kappa_t)\, d\kappa_n \tag{4}$$

In the above set of equations, (2) is by Bayes' rule, (3) is due to the marginalization over κ_n and κ_t, (4) is simply the factorization of a joint probability, and (5) uses the assumption that ρ, κ_n and κ_t are independent of each other and of N.

In our implementation, we normalize this function numerically by calculating the probability of 100 equally spaced values for rho in [0,1]. We report the mode of $P(\rho)$ as the purity estimate, along with 95% confidence intervals, calculated as the values corresponding to 2.5% and 97.5% of the cumulative distribution.

2.4.8 Sample clustering approach to labeling normal and abnormal CD138+ cells

For each sample, we performed a cluster analysis of only the cells in that sample. More specifically, we calculated variable genes based on log normalized expression using the `highly_variable_genes` function from Scanpy (version 1.7.1) [72] with parameter `min_disp=0.6`, centered and scaled the data, clipping the resulting values to a maximum of 10, and ran PCA and Leiden clustering (determining the number of PCs to input to Leiden clustering based on an elbow plot). We manually inspected the resulting clusters for each sample to determine whether each cluster contained healthy or abnormal cells. This determination was based on whether the cluster uniquely expressed the clonal immunoglobulin for that tumor [73] (since immunoglobulins were removed from the highly variable gene list, they did not influence the clustering results), as well as each cluster's expression of certain oncogenes, such as *CCND1* for t(11;14) tumors. All cells were labeled in this way, except for 20 cells from sample MGUS-2 that were characterized by low expression of *MALAT1* and were not obviously similar to the healthy or abnormal cells from that sample, and thus were not classified and were excluded from downstream analyses. See Figure 2.4 for an example of this

method applied to a sample.

In addition to labeling each individual cell as normal or abnormal, this approach allowed us to determine a tumor purity for each sample, i.e. the fraction of cells labeled abnormal. To calculate confidence intervals on this purity estimate, we assumed that our observed data was generated as $n \sim \text{Binomial}(N, p)$, where N is the total sequenced cells in a sample, n of which we labeled as abnormal, and p are the proportion of abnormal cells in the patient sample (not just the ones we sequenced). We further assume a uniform prior on p ($p \sim \text{Beta}(1, 1)$), thus the posterior distribution on p is $p|n, N \sim \text{Beta}(n + 1, Nn + 1)$, by conjugacy of the Beta and Binomial distributions. We derived 95% confidence intervals on each sample's purity estimate based on the inverse cdf of its beta-distributed posterior.

2.4.9 Abnormal vs. normal differential expression testing with limma

DEGs between abnormal and normal cells were derived using limma version 3.42.2 with voom transformation [35, 36, 74]. Samples were split into their abnormal and normal populations, and we refer to each of these as a "pseudosample." Counts across cells in a pseudosample were summed and used as input to the limma-voom pipeline. Immunoglobulin genes and genes with counts per million (CPM) <5 in all samples or expressed in <5% of both abnormal and normal cells were removed prior to analysis, resulting in normalization and DE testing of 6,521 genes. Pseudosamples were normalized using the trimmed mean of M values (TMM) method61 and fold changes were calculated as implemented in limma. We controlled for age, sex, sample preparation batch, and whether the sample was fresh or frozen. Age information was missing for one NBM sample, and we filled it using mean imputation based on the ages of the other NBM samples. DEGs were those with a Benjamini-Hochberg FDR<0.1 and |log fold change| > log(1.5).

2.4.10 Within-patient differential expression testing

For each patient with both abnormal and normal cells detected, we calculated DEGs between their abnormal and normal cell populations using a Wilcoxon rank sum test, correcting for multiple hypothesis testing across genes tested for each patient. We calculated fold changes as implemented in Scanpy (version 1.7.1) [72], but replacing their offset term of 1×10^{-9} with half of the minimum (non-zero) log-normalized expression value in our data (0.126), to avoid inflating fold changes. Specifically, fold change was calculated as the ratio of

$$\exp[\frac{1}{N}\sum(\text{log-normalized expression}) - 1] + \text{offset}$$

in each group, where N denotes the number of cells in the group. Differentially expressed genes were those with a Benjamini-Hochberg FDR<0.1 and |log fold change| > log(1.5).

For visualizing DEG uniquely found using this within-patient DE approach (Figure 2.9), we first limited DEG to those not found using limma. Then, for each gene, we calculated a maximum log2(q-value) as the maximum (BH-corrected) q-value reported across patients, multiplied by the number of patients with DEG (10, in our data) to further correct for multiple hypothesis testing across patients. This value was calculated separately for upregulated and downregulated instances of DEG, where applicable.

2.4.11 Automatic relevance determination nonnegative matrix factorization (ARD-NMF)-derived gene expression signatures

We defined gene expression signatures using our SignatureAnalyzer-GPU tool [53] (see Code Availability), which implements a previously described ARD-NMF algorithm [6]. This method approximates the gene expression profile of each cell (represented as a column in the genes-by-cells input matrix, V) as an additive combination of latent gene expression signatures (each column in the genes-by-signatures W-matrix), each with an associated weight or 'activity' in each cell given by the signatures-by-cells H-matrix:

$$\mathbf{V} \approx \hat{\mathbf{V}} \triangleq \mathbf{WH}$$

This Bayesian variant of NMF encourages sparse interpretable solutions by imposing either exponential or half normal priors on the weights of the W- and H-matrices and allows automatic discovery of the number of signatures (K) required to explain the data. It solves for the W and H matrices using maximum a posteriori (MAP) estimation over

$$p(\mathbf{W}, \mathbf{H}, \lambda | \mathbf{V})$$

, where λ is a vector of signature relevance weights. Using a Poisson noise model for our data, an exponential prior on W, and a half-norm prior on H, the objective function for ARD-NMF described in equation 19 of the original paper [6] is given by:

$$-\log p(\mathbf{W}, \mathbf{H}, \lambda | \mathbf{V}) = D_{KL}(\mathbf{V} | \mathbf{W}, \mathbf{H}) + \sum_{k=1}^{K'} \frac{1}{\lambda_k} (\sum_{g=1}^{G} w_{g,k} + \sum_{c=1}^{C} \frac{h_{k,c}^2}{2} + b)$$
$$+ (G + \frac{C}{2} + a + 1) \log \lambda_k + cst(a, b)$$

where g represents a given gene out of G total genes, c represents a given cell out of C

total cells, a and b are hyperparameters (how we chose a and b is described below), λ_k is a learned relevance weight for signature k, w and h represent elements from the W and H matrices respectively, and const(a,b) is a constant that depends only on a and b.

After signature discovery, the columns of W were normalized to a sum of 1 and all the weight was shifted into the H-matrix:

$$\mathbf{W}_{g,k} \leftarrow \frac{\mathbf{W}_{g,k}}{\sum_{g'=1}^{G} \mathbf{W}_{g',k}}$$

$$\mathbf{H}_{k,c} \leftarrow (\sum_{g=1}^{G} \mathbf{W}_{g,k})\mathbf{H}_{k,c}$$

for gene g (out of G total genes), signature k and cell c.

Our input data was UMI counts for 3,883 highly variable genes (dimensions of V = 3,883 x 29,387), which were determined using the `highly_variable_genes` function from Scanpy (version 1.7.1) [72] with `min_disp=0.2`, which we set to be lower than the default value so as to include genes which may have relevance to plasma and myeloma cell biology despite a modest dispersion value. In addition to other default settings for the SignatureAnalyzer-GPU tool, we use a Poisson objective with an L1 prior on W and an L2 prior on H, set the initial K to 50, the maximum number of iterations to 7,000, and the tolerance to 110-5. Following the guidelines in the original ARD-NMF paper [6, 53], we set hyperparameter a = 10 (the default in SignatureAnalyzer-GPU) and then calculate b as a function of a, as implemented by SignatureAnalyzer-GPU. We held out 20% of cells as a validation set. Since the Bayesian NMF algorithm finds a local minimum each time it is run, we ran the algorithm 100 times on our data in order to choose an optimal solution. Over 100 runs, the algorithm returned solutions with K between 24 and 30 with a mode of 28, and we chose the set of signatures with the lowest beta divergence over the validation set from among the solutions with K = 28 (dimensions of W = 3,883 x 28; H = 28 x 29,387). Before analyzing the signature results, we normalize each column in H by that cell's total counts.

A signature was classified as "patient-specific" if its mean activity across cells in any one patient is >4 standard deviations higher than in all other patients. Otherwise, a signature is classified as "single-gene" if the weight of its most highly weighted gene based on the W matrix is ≥ 0.5 more than the weight of its next highest weighted gene. If a signature doesn't meet either of these criteria, we describe it according to its top genes, where signature genes are ranked by their weight in W multiplied by their specificity to that signature, with specificity S defined as:

$$S_{g,k} = \frac{[\mathbf{H} \cdot \mathbf{1}]_k \mathbf{W}_{g,k}}{\sum_{k'=1}^{k'} ([\mathbf{H} \cdot \mathbf{1}]_{k'} \mathbf{W}_{g,k'})}$$

Signatures significantly altered between disease states were identified by calculating the mean signature activity for the abnormal and normal cell populations in each sample, respectively, and performing a Kruskal-Wallis one-way analysis of variance and Dunn's multiple comparison test with Bonferroni correction to detect differences in mean activities between the following groups: NBM, normal MGUS, normal SMM, normal MM, abnormal MGUS, abnormal SMM, and abnormal MM. Comparisons with family-wise error rate < 0.1 were considered significant.

We additionally ran Bayesian NMF on an external single cell dataset [28] using the same methods as above. We first limited the external data to cell types and disease stages which are present in our data, retaining only bone marrow PCs derived from healthy donors and patients with MGUS, SMM or MM, and limited the input features to hypervariable genes across these cells (4,669 genes).

2.4.12 Testing to ensure that signature activity did not correlate with batch variables

For our NMF analysis, we interrogated whether the activity level of any signature is correlated with batch variables (age, sex, sample preparation batch, and fresh/frozen storage). To do this, we limited our data to normal plasma cells, for which we would not expect to find significant differences in signature activity between samples, and then tested for differences in distributions of signature activities between samples from the different batch groups (testing each batch variable separately; using a rank sum test for sex and fresh/frozen, a Kruskall Wallis rank test for batch, and Pearson and Spearman correlations for age). None of the batch variables were significantly correlated with signature activity, using $p<0.05$ as the significance threshold. Given that we did not observe batch-related differences in signature activity in our normal plasma cells, we conclude that the differences that we observed between abnormal samples at different disease stages are indeed driven by the disease.

2.4.13 Estimating normal plasma cell signature activity in the MMRF dataset

To estimate the activity of a gene expression signature for each sample in the publicly available MMRF bulk RNA-sequencing dataset (https://research.themmrf.org), we: calculated log-normalized transcripts per million (tpm) on the MMRF counts data using the DESeq2 method for size factors, where samples are normalized using the median, across genes, of the ratios of gene counts to each gene's geometric mean across samples62. Then, for top signature genes (for our normal plasma cell signature, these included *CD27, CD79A, RNU12, JSRP1, SAT1,*

CTSH, and *HCST*), we z-scored the log-tpm expression of each gene across samples, and calculated the signature activity as the mean of z-scored gene expression values.

2.4.14 Pseudobulking procedure

To pseudobulk samples, we summed the gene counts across cells, calculated the total gene counts in the sample (ignoring genes that accounted for >5% of counts), divided the summed count vector by the total gene counts, and multiplied by one million.

2.4.15 Single sample GSEA (ssGSEA)

Samples were split into their abnormal and normal populations, and we refer to each of these as a "pseudosample." We calculated the pseudobulk expression for each pseudosample and input this to the ssGSEA module available on the GenePattern platform [75, 76] to calculate enrichment scores for the hallmark gene sets provided by the Molecular Signature Database (MSigDB) [59]. We removed pseudosamples comprised of <20 cells from downstream analysis of ssGSEA results, due to the high variance inherent in their gene expression. Differential pathway activity between two groups of pseudosamples was calculated using a t-test, and pathways with BH FDR < 0.1 were reported.

2.4.16 Assessing intratumor heterogeneity for NMF signatures

For each sample, we limited our analysis to abnormal cells and used Scanpy's (version 1.7.1) [72] built-in functions to compute highly variable genes (`min_mean=0.0125`, `max_mean=3`, `min_disp=0.6`; genes located on immunoglobulin loci were removed), scale the data (`max_value=10`), compute the 10 first principal components, compute a neighborhood graph (`n_neighbors=15`) and run Leiden clustering (resolution=0.6). Parameters that differed from the Scanpy defaults are shown in parentheses. This defined clusters for each sample. We determined that a sample contained a heterogeneous population of cells vis-a-vis a given signature if the mean activity of that signature had a coefficient of variation > 1 across clusters. Specifically, for a given sample, we calculated the mean activity of a given signature in each cluster, producing a vector of means μ with length equal to the number of clusters. We then considered a sample-signature pair to exhibit intratumor heterogeneity if $\frac{\text{std}(\mu)}{\text{mean}(\mu)} > 1$. Signature activities across sample clusters are shown in Figure 2.24 for all signature-sample pairs which passed this threshold.

2.4.17 Statistical analysis

Kruskal-Wallis one-way analysis of variance and Dunn's multiple comparison test with Bonferroni corrections were used when three or more independent groups were compared. When comparing two independent groups, all parametric tests were two-tailed, and the Benjamini-Hochberg (BH) method was used to correct for multiple hypothesis testing where appropriate. $P < 0.05$ or $q < 0.1$ (in cases of multiple hypothesis correction) were considered statistically significant. Error bars plotted on visualizations of mean signature activity or gene expression in a sample represent the standard error of the mean, and were calculated as the standard deviation of the means of 10,000 bootstrapped versions of that sample.

2.5 Data Availability

The scRNA-seq data generated in this study have been deposited in the NCBI Gene Expression Omnibus (GEO) database under accession number GSE193531 [https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE193531]. Raw data have been deposited in dbGaP under accession number phs001323.v3.p1 [https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs001323.v3.p1]. The publicly available MSigDB gene sets used in this study are available from the Molecular Signatures Database v7.5.1 [http://www.gsea-msigdb.org/gsea/msigdb]. MMRF CoMMpass data used in this study can be obtained from the MMRF Research Gateway [https://research.themmrf.org].

2.6 Code Availability

Analysis code is publicly available at https://github.com/getzlab/Boiarsky-etal-2022.

Chapter 3

Overcoming Copy Number Effects in Tumor Cell Embeddings

3.1 Introduction

Elucidating elements of cancer biology that are common between different cancer types and subtypes is of great interest to the oncology community. Much progress has been made in identifying recurrent genetic drivers of tumor initiation and progression [77], but shared cellular phenotypes have been harder to pin down due to the vastly different transcriptional profiles between different cancers. Moreover, even identifying cells with similar phenotypes among multiple patients with the same cancer type can be laborious owing to the widescale transcriptional differences between tumors from different patients. We observed these phenomena firsthand in Chapter 2, where we required advanced computational tools like Non-Negative Matrix Factorization to discover shared gene signatures across multiple myeloma patients (Figure 2.14), and a laborious per-patient cluster analysis to identify abnormal plasma cells (Figure 2.6), as each patient's cancer cells were so transcriptionally distinct from one another.

Dimensionality reduction and clustering are prototypical parts of single cell RNA-sequencing (scRNA-seq) analysis and are often used to elucidate and visualize groups of cells that share gene expression programs and as a precursor before calculating differential expression of genes between cell clusters [72, 78]. In cancer, this step of the workflow is often confounded by overwhelming differences between the transcriptional profiles of tumors from different patients, obscuring our ability to use standard pipeline tools such as principal component analysis (PCA) and t-stochastic neighbor embedding (tSNE) to embed cells in a latent space that relates information about cell state and disease status. Instead, cells often separate by patient in the latent space, obscuring cell subtypes and gene expression

activities which may be shared across patients — another phenomenon observed in Chapter 2 [30, 79–81].

As somatic copy number events are a common occurrence in tumor samples, we hypothesized that distinct copy number variations (CNVs) across patients drive these large-scale transcriptional differences between tumors. In this chapter, we investigate the effect of CNVs on gene expression and explore approaches for correcting for copy number differences between tumors when embedding cells into a latent space. We aim to generate discussion about the role of CNVs in driving distinct transcriptional profiles between tumors, and the efficacy and pitfalls of different computational approaches for correcting for CNVs when embedding single cells in the context of cancer.

3.2 Related work

3.2.1 Effects of somatic CNV on gene expression

Somatic copy number variations (CNVs) are widespread genomic alterations in cancer, resulting in gains or losses of large DNA segments and affecting hundreds to thousands of genes [82]. These alterations can influence cancer biology both by amplifying oncogenes and deleting tumor suppressor genes, and by modulating gene expression programs more broadly. The presence of CNVs and their roles in tumorigenesis have been well established [83–85].

In bulk RNA-seq datasets, numerous studies have demonstrated that copy number amplifications tend to result in increased mRNA levels of genes located in the amplified regions, and conversely, deletions often lead to reduced expression [86]. However, single-cell RNA-seq studies have begun to reveal a more nuanced picture. At the single-cell level, CNVs do not uniformly lead to proportional changes in gene expression. Instead, the transcriptional consequences of CNVs may depend on a combination of factors including chromatin context, gene regulatory networks, epigenetic state, and cell type identity [87]. Recent works such as inferCNV [**taylor2021single**, 87] and CopyKAT [**gill2021copykat**] have proposed computational methods to infer CNVs directly from scRNA-seq data, relying on changes in patterns of gene expression across chromosomal regions.

Importantly, several studies have begun to quantify the extent to which gene expression variability in cancer can be explained by CNV burden. Bhattacharya et al. [88] analyzed pan-cancer data from The Cancer Genome Atlas (TCGA) and demonstrated that a significant proportion of variance in gene expression across tumors is attributable to CNVs, with the strongest effects observed in genes located within CNV regions. Shao et al. [86] further emphasized that expression-CNV correlation strength varies by gene, suggesting that post-

transcriptional mechanisms or dosage compensation may buffer expression in some cases.

Understanding the effect of CNVs on gene expression is especially critical in single-cell studies of cancer, where intra-tumor heterogeneity and patient-specific CNV landscapes may introduce major confounding signals in dimensionality reduction and clustering. Consequently, efforts to correct or model the effects of CNVs are increasingly important when trying to identify shared cell states or gene expression programs across patients with distinct CNV profiles.

3.2.2 Batch correction for single cell RNA-seq data

Single-cell RNA sequencing (scRNA-seq) datasets often suffer from batch effects — systematic sources of variation that arise from differences in sample processing, library preparation, sequencing platform, and other technical factors. These effects can obscure the true biological signal and hinder downstream analyses such as clustering, trajectory inference, and differential expression [89]. Batch effects are especially problematic when integrating scRNA-seq data from multiple patients, time points, or experimental conditions, and can be easily mistaken for biological differences if not properly addressed.

To overcome these challenges, a range of computational batch correction and integration methods have been developed. These approaches aim to project cells into a shared latent space where the technical sources of variation are minimized while preserving meaningful biological differences. For example, Seurat v3/v4 [90, 91] identifies shared features between datasets (termed "anchors") using canonical correlation analysis (CCA) and mutual nearest neighbors (MNNs), and uses these to integrate cells across batches. Seurat has become a standard tool for integrating data from different patients, conditions, or modalities. Harmony [92] projects cells into a shared low-dimensional embedding and iteratively corrects for batch variables using soft clustering and linear regression. It is fast, scalable to large datasets, and integrates directly into PCA-based workflows. Single-cell Variational Inference (scVI) [12] uses a deep generative model to learn a latent representation of gene expression that is disentangled from known batch variables. scVI is particularly powerful when batches correspond to known covariates (e.g., patient ID, donor, library) and is widely used for probabilistic modeling of single-cell data. scANVI [93] is a semi-supervised extension of scVI that allows for partial labeling of cells, improving transfer of cell type annotations across batches and conditions. Finally, scPhere [94] uses spherical autoencoder that projects cells onto a hypersphere in latent space and corrects for batch effects during the projection. scPhere has been shown to preserve global data structures and outperform several existing methods in trajectory and integration tasks.

While these methods perform well for many integration tasks, they may fall short in cancer studies where transcriptional variability is driven not only by technical noise but also by genuine biological differences such as somatic mutations, CNVs, or tumor microenvironment heterogeneity. Standard batch correction methods may inadvertently remove or distort signals related to disease biology if these are confounded with batch labels (e.g., patient ID), leading to overcorrection and loss of relevant variation. This is particularly problematic in cancer, where each patient's tumor may have a unique molecular signature that is both biologically meaningful and aligned with the batch variable.

Consequently, careful interpretation of batch correction outputs is essential in oncology settings. In some cases, it may be preferable to use methods that explicitly model shared vs. private sources of variation (as discussed in the next section), or to stratify the analysis to avoid overcorrecting meaningful patient-specific signals.

3.2.3 Identifying latents that are shared vs. unique between samples

In complex biological systems, particularly in cancer, not all variation between samples should be considered technical noise. Tumors from different patients can exhibit genuine differences in gene expression patterns due to variations in genetic alterations, tumor microenvironments, or cellular composition. Traditional batch correction methods, which aim to minimize between-sample variation, may inadvertently remove biologically meaningful signals when applied too aggressively. An alternative approach is to explicitly model and disentangle the sources of variation across samples, identifying latent factors that are either shared across all samples or unique to specific subsets.

Several computational frameworks have been developed with this goal in mind. These approaches generally fall into two methodological categories: matrix factorization-based and autoencoder-based models.

Two notable matrix factorization approaches are LIGER and DIALOGUE. LIGER (Linked Inference of Genomic Experimental Relationships) [95] uses integrative non-negative matrix factorization (iNMF) to decompose gene expression matrices from different samples into shared and dataset-specific factors. This allows the model to capture latent programs that are common across patients while also preserving heterogeneity that may be biologically meaningful. LIGER has been successfully applied to datasets with substantial variation, including human brain and tumor tissue, where patient-specific effects are prominent. DIALOGUE [96] extends the idea of shared vs. unique latent factors to a supervised context. It models gene programs that are associated with specific phenotypes or clinical traits while accounting for confounders. Although originally designed to identify intercellular signaling programs

across cell types, the framework also supports the decomposition of transcriptional variance into shared and private axes.

Autoencoder-based approaches include multiVI, multiGroupVI, and MrVI, among others. multiVI [97] and multiGroupVI [98] are probabilistic models that use variational autoencoders (VAEs) to jointly model multiple sources of variation in single-cell data. These methods aim to disentangle latent factors that are shared across conditions or patients from those that are sample-specific. Importantly, they allow users to specify or infer group-level structure (e.g., patients, tissues) during training, which improves interpretability and enables more robust transfer of information across datasets. MrVI (Multi-resolution Variational Inference) [99] is a hierarchical model that simultaneously learns two latent spaces: one that is "aware" of group-specific variation (e.g., batch effects, CNVs), and one that is "unaware" of such variables. This separation offers a powerful tool in cancer studies, where users may want to analyze tumors either in a CNV-sensitive embedding (to examine relationships between transcription and genomic alterations) or in a CNV-corrected embedding (to discover shared phenotypes across patients). MrVI also enables exploration of hierarchical or nested variation — an increasingly relevant paradigm in tumor biology where variation exists at the level of patient, tumor, clone, and cell.

This class of methods offers an attractive solution in the cancer setting: rather than attempting to remove between-patient variation entirely, they allow for the controlled exploration of which features are shared and which are patient- or clone-specific. For instance, in our own work on multiple myeloma (see Chapter 2), we found that NMF-based discovery of gene expression programs revealed reproducible, interpretable signatures of plasma cell states across patients, while per-patient clustering captured unique malignant subpopulations. This suggests that a multi-resolution perspective is critical for understanding tumor heterogeneity.

Together, these methods provide powerful alternatives to traditional batch correction, enabling researchers to preserve and interrogate biologically meaningful variation across diverse samples — a key goal when studying transcriptional phenotypes in cancer.

3.3 Experiments and Results

3.3.1 Datasets

We worked with three clinical single cell RNA-sequencing datasets from cancer:

- The multiple myeloma (MM) dataset [30] contained 29,387 plasma cells (putative cancer cell of origin) and myeloma cells from 26 patients at varying disease stages and 9 healthy donors.

- The synovial sarcoma (SyS) dataset [100] contained 16,872 malignant, immune, and stromal cells from 12 human SyS tumors.

- The colorectal cancer (CrC) dataset [101] contained 21,657 malignant, immune, and stromal cells from 15 human CrC tumors. We used the version of the [101] CrC dataset that was preprocessed by [81] and available on the Curated Cancer Cell Atlas (3CA; https://weizmann.ac.il/sites/3CA). As Gavish et al. note, this dataset contains 15 colorectal carcinoma samples from the SMC cohort, sequenced at the Samsung Medical Center in Seoul. The full cohort consists of 23 tumor samples – 8 were excluded due to insufficient CNA signal. Cells with fewer than 1000 genes detected were also excluded.

3.3.2 Characterizing the effect of copy number on single cell gene expression

In order to estimate copy number profiles from single cell RNA-sequencing data, we ran inferCNV [102] on the MM and SyS datasets, and used the results of 'inferCNA' provided by the authors of [81] for the CrC dataset. The output from these methods were gene level estimates of relative copy number change in each cell.

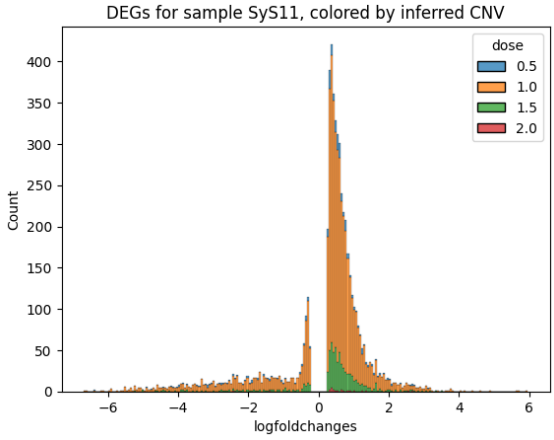
We sought to understand the role of copy number in driving gene expression differences between tumors. Given the estimated CNV profiles of each cell, we interrogated whether genes that are uniquely expressed in individual tumors were involved in copy number changes more often than they would be by chance.

To calculate differentially expressed genes (DEGs) between different tumors, we used a Wilcoxon rank sum test to compare the log-normalized gene expression in different tumors. Any gene with a Benjamini-Hochberg adjusted p-value < 0.1 was retained as a DEG. Fold changes were calculated as the ratio of log-normalized expression + offset for cells in a given tumor, to that of cells in other tumors. The offset used is half of the non-zero minimum of log-normalized gene expression in the dataset. For datasets that contained non-malignant cells (eg. the CrC and Sys datasets), we had an additional post-processing step where we calculated patient-specific DEGs for all other annotated cell types (eg. T cells from patient A vs. T cells not from patient A), and removed these non-malignant patient specific genes from our list of malignant DEGs, such that the malignant DEGs reflect differences that arise specifically between tumors, not simply between patients. For example, we saw the sex gene *XIST* removed from the tumor DEG list after post-processing in this way.

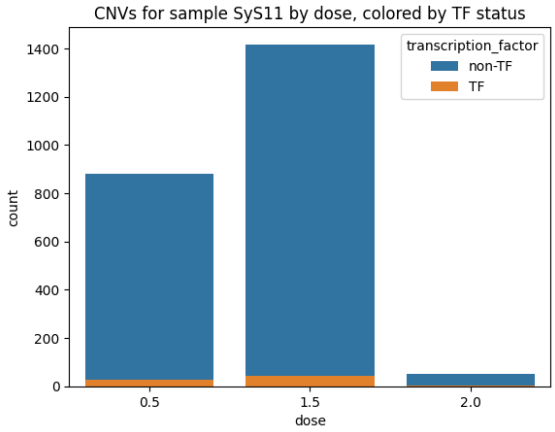
Taking the SyS dataset as an example, we calculated differentially expressed genes (DEGs) between the cells in a given tumor vs. all other tumors. Each tumor had between 491 and 7,224 DEGs, with a mean of 4,889 DEGs per tumor. Of the DEGs, on average 20.5% were

involved in a CNA (range [8.5, 40.3]). These numbers reveal that the majority of genes unique to a given tumor are not directly involved in a copy number event (Figure 3.1a).

While it seemed that gene dosage change as a direct result of deletion or duplication does not account for most differential expression between tumors, we next investigated whether CNVs might still have a major influence on the unique gene profiles of each tumor through 1) exceptionally large fold changes for DEGs that were involved in CNVs compared to those that were not or 2) copy number events involving transcription factors driving differential expression of many downstream target genes. For 9 out of 12 tumors in the SyS dataset, the fold changes of DEGs involved in a CNA were not significantly more extreme than those of DEGs not involved in a CNA (Wilcoxon rank sum test). We downloaded a list of 795 transcription factors from the JASPAR database [103], 617 of which overlapped with the gene names in the SyS dataset. For a given tumor, most of the genes affected by CNAs were not transcription factors (Figure 3.1b).



(a) For representative sample Sys11, a histogram of the log fold changes for differentially expressed genes, with each DEG colored by its inferred CNA dosage.



(b) For representative sample SyS11, a histogram of the inferred dosage change for genes with copy number alterations, colored by whether or not the gene is listed as a transcription factor in the JASPAR database.

Figure 3.1: Relationship between DEGs and CNVs in representative Synovial Sarcoma sample.

3.3.3 Generative model for copy number correction

Given our results in section 3.3.2 which showed that the direct gene dosage effect of CNVs does not account for the majority of transcriptional differences between tumors, we explore correcting for CNVs in a non-linear fashion. We use the scvi-tools framework [104], which allows for correcting continuous covariates associated with each cell when embedding cells

to a latent space using a variational autoencoder. We compared the results of 3 different generative models, shown in Figure 3.2.

The first generative model was scVI, proposed by Lopez et al. (2018) (Figure 3.2b). Building off of vanilla scVI, we experiment with two batch correction schemes. We hypothesize that while embedding cells without any correction (Figure 3.2b) clusters cells from different patients separately in the latent space, performing batch correction based on patient ID (PID) (Figure 3.2c) may be too blunt a correction, obscuring meaningful biological differences between patients. We propose correcting for CNV profiles (Figure 3.2a), which may overcome a significant portion of transcriptional differences between patients in a principled manner, namely correcting for the effects of CNVs while maintaining other meaningful patient differences (e.g. disease subtypes). To this end, we used principal component analysis (PCA) to reduce the dimensionality of each cell's CNV profile from g genes to 20 dimensions, and then retained the top 5 principal components (PCs) that explained the most variance among CNV profiles, and input this vector as a per-cell batch variable.

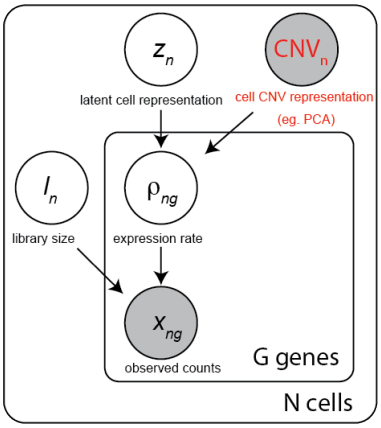
Our proposed generative model that corrects for CNVs within the scVI framework (corresponding to Figure 3.2a) can be formalized as follows: given scRNA-seq counts x_{ng} and copy number information c_{ng} for each gene g in each cell n (which can be estimated from the RNA-seq data itself using software like InferCNV)

$$z_n \sim \text{Normal}\,(0, I) \tag{3.1}$$
$$\ell_n \sim \text{log normal}\left(\ell_\mu, \ell_\sigma^2\right) \tag{3.2}$$
$$\rho_n = f_w\left(z_n, c_{ng}\right) \tag{3.3}$$
$$w_{ng} \sim \text{Gamma}\left(\rho_n^g, \theta\right) \tag{3.4}$$
$$x_{ng} \sim \text{Poisson}\left(\ell_n w_{ng}\right) \tag{3.5}$$

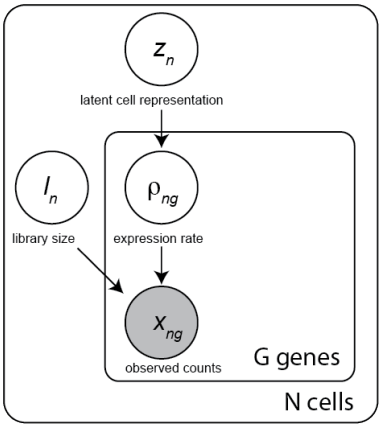
where l_n represents the library size (or total counts) in each cell, f_w is a function parameterized by a neural network, and $\theta \in \mathbb{R}_+^G$ denotes a gene-specific inverse dispersion, estimated via variational Bayesian inference.

3.3.4 Running scVI

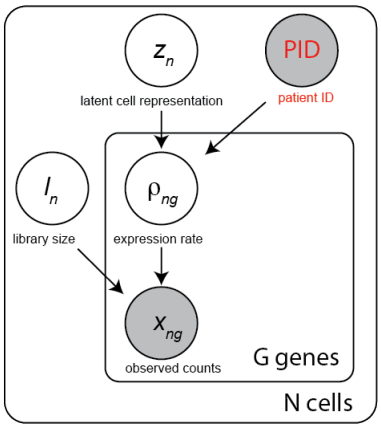
Prior to running scVI, each dataset was limited to 5,000 highly variable genes (HVGs), in order to speed up training times in scVI. HVGs were chosen using the function `scanpy.pp.highly_variable_genes()` with the argument `flavor="seurat_v3"`. scVI was run as implemented in the scvi-tools package version 1.0.0. To correct for CNV, we called the



(a) A plate representation of the generative model used when correcting for CNA profiles as a continuous covariate.



(b) A plate representation of the generative model used for scVI without any batch correction.



(c) A plate representation of the generative model used when correcting for patient ID as a continuous covariate.

Figure 3.2: In our main set of experiments, we compared the results of three generative models of gene expression, each represented above. Red text emphasizes the variable that was corrected for in a given model.

`scvi.model.SCVI.setup_anndata` function with the argument `continuous_covariate_keys` set to the PCA-based CNV features in the anndata .obs matrix, and to correct for PID, we called the same function with the argument `batch_key` set to the column in the anndata .obs matrix containing patient identifiers. We opted to model the likelihood of gene expression counts using the negative binomial distribution, as implemented in scvi-tools.

3.3.5 Evaluation setup

In evaluating the effect of each correction scheme (no correction, correcting for PID, or correcting for CNV), we assessed the resulting latent representation of each cell using a few metrics:

1. The extent to which the patient ID (PID) was present in the latent space, quantified by the average accuracy for a logistic regression model predicting which patient a cell originated from given its latent embedding. We expect this value to be highest in the latent space with no batch correction, lowest in the latent space that corrected for PID, and intermediate in the latent space that corrected for CNV, since CNV profiles are patient-specific.

2. The extent to which cells clustered according to meaningful biology in the latent space. For this metric, we used the 41 gene meta-programs (MPs) defined in recent work

that studied transcriptional heterogeneity across 1,000 tumors [81], and assessed how spatially autocorrelated each MP was in the latent space using Geary's C value, which has previously been applied to assess single cell similarity [105, 106]. Briefly, Geary's C is calculated as

$$C = \frac{(N-1)\sum_i \sum_j w_{ij}(x_i - x_j)^2}{2W\sum_i (x_i - \overline{x})^2}$$

where N denotes the total number of cells, w_{ij} refers to the connectivity weight between cells i and j in a KNN neighborhood graph, x represents a value of interest, in our case the level of gene MP activity, W is the sum of all weights, and $\overline{x}$ is the mean of x. Following [106], we report $C^* = 1 - C$, such that scores generally range between 0 and 1, with 0 representing no correlation, and 1 representing high correlation.

3.3.6 Experimental results

We ran the three scVI models described in Figure 3.2 each five times, in order to capture the variation in latent space resulting from the stochasticity of the model and learning procedure. We found that results were qualitatively similar across the five runs for each model. In Figure 3.3, we share unified manifold approximation and projection (UMAP) plots representing the latent space learned using each model for the multiple myeloma dataset, coloring cells by patient ID and disease stage in order to aid in tracking their relative positions under the three different models. We then evaluated each latent space according to the metrics described above.

Presence of PID in the latent space.

As expected, across all datasets, PID is least predictable from the model which explicitly regresses out the PID. Further, the fact that PID is still predictable from the latent spaces corrected for CNV reflects that while CNV profiles are correlated with PID, they contain only a subset of patient-specific information. We note that as we increase the number of CNV PCs that we use in the correction, the accuracy of predicting PIDs from the CNV-corrected latent space further decreases, reflecting the fact that the CNV principal components may in fact be capturing patient information, but that the granularity of the correction can be tuned by choosing how many PCs to include in the correction.

Spatial autocorrelation of gene MPs in the latent space.

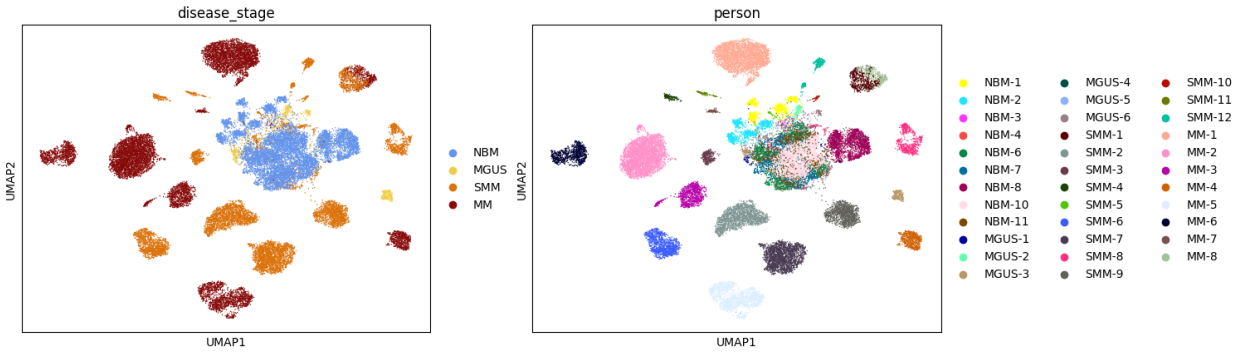
In Figure 3.4, we present results for the spatial autocorrelation of some representative MPs in the MM dataset across the three different latent spaces learned (each latent space was run

with five different seeds to capture stochasticity; for full results). In particular, we present the six MPs that had the highest mean C^* value in the uncorrected latent space, as these are gene programs with high activity in the MM dataset. We find that while the spatial autocorrelation structure is lost when correcting for PID, it is retained in the latents that instead corrected for CNVs.

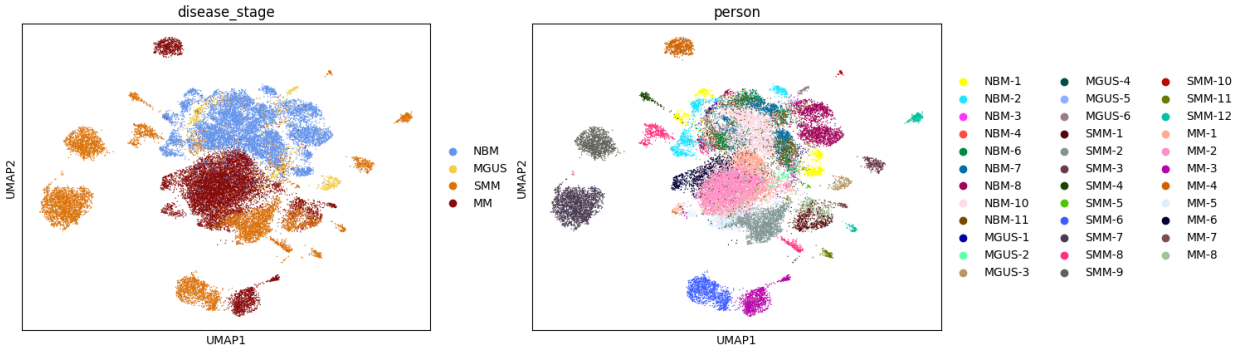
3.4 Discussion

While it is believed that CNVs play a large part in driving the wide-scale transcriptional differences between tumors from different patients, there is limited research characterizing the extent of their influence on transcription relative to other factors that can affect transcription, such as point mutations, cytogenetic changes, and influences from the microenvironment. Additionally, the effect of CNVs on transcription may vary between different cancer types or subtypes. There have been some reports of correlation between disease biology and CNVs, and therefore it is possible that by regressing out CNVs from a latent cell representation, one will also regress out important aspects of disease biology. For example, [81] reports multiple correlations between specific CNVs and expression of specific gene meta-programs, and these associations vary between different cancer types. In medulloblastoma, [107] report correlations between specific CNVs and cell proliferation as well as the developmental stage of the malignant cells, respectively. An undesirable effect of regressing out CNVs may be regressing out important cellular phenotypes, such as these, and we therefore advise caution in implementing this kind of approach.

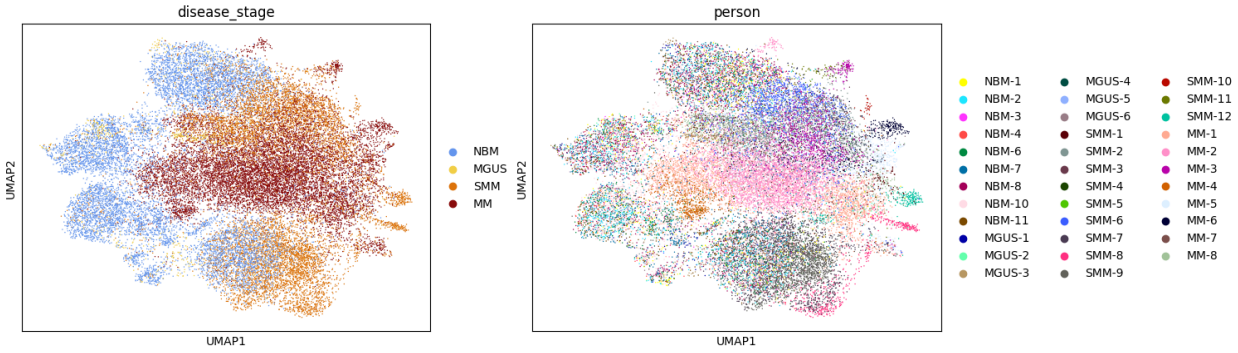
In the form we presented here, correcting for CNVs represents an intermediate correction between no correction at all, and correcting for each patient individually. One framing of our proposed method is that CNV profiles meaningfully *group* patients, and thus regressing out this information brings groups of patients closer together while maintaining some inter-patient structure in the learned latent space. In practice, with the small amount of data used here, it is not clear that the patient groupings are biologically meaningful, and future work is needed to understand how CNVs vary between patients and how this variation can be used to group patients together before doing a batch correction.



(a) UMAP of cells in latent space after running scVI with no correction.



(b) UMAP of cells in latent space after running scVI and correcting for CNAs, using the first seven principal components.



(c) UMAP of cells in latent space after running scVI and correcting for patient ID.

Figure 3.3: UMAP plots of the latent space learned using three different correction schemes for the multiple myeloma dataset. Cells are colored by disease stage (left; NBM=normal bone marrow, MGUS=monoclonal gammopathy of undetermined significance, SMM=smoldering multiple myeloma, MM=myeloma) and patient ID (right).

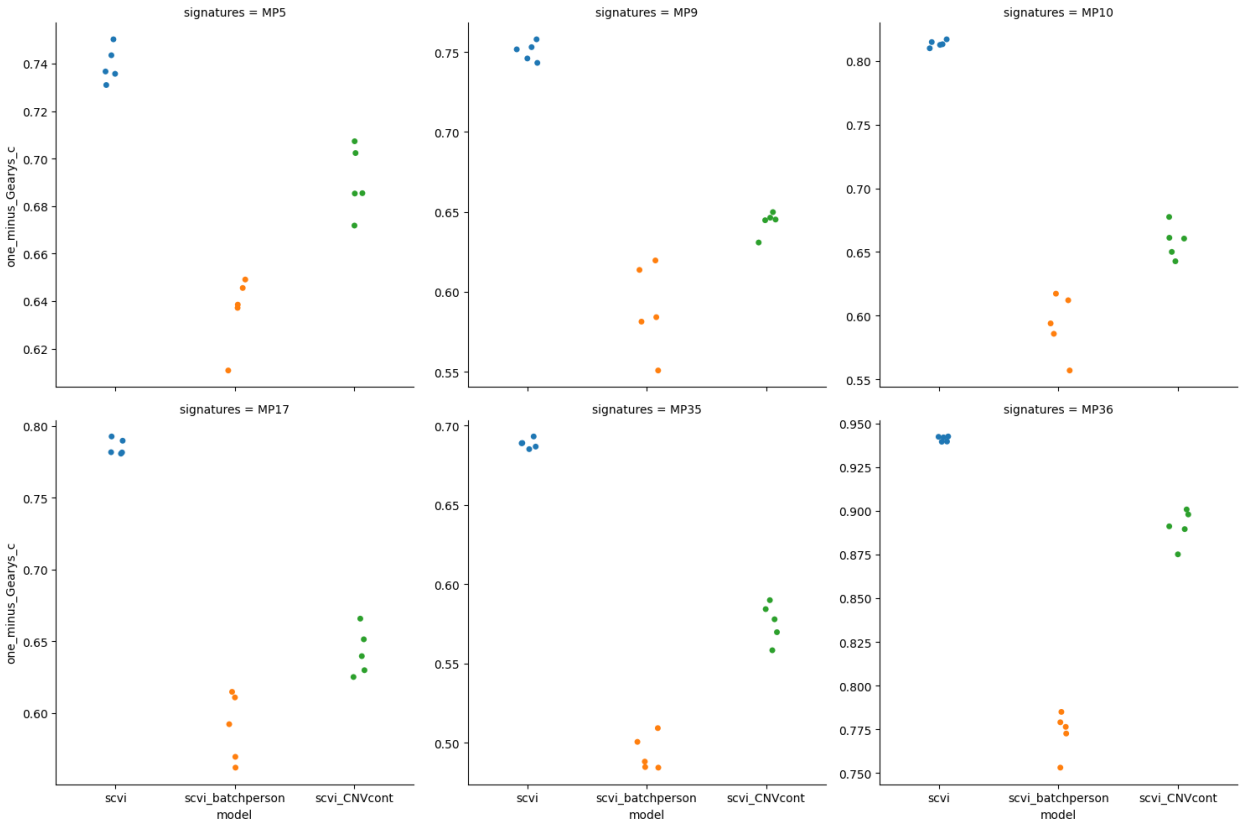


Figure 3.4: For the MM dataset, C^* (1-Geary's C) for the six gene meta-programs (MPs) with the highest mean activity in the uncorrected latent space (mean taken across 5 runs of scVI). We find that while the spatial autocorrelation structure is lost when correcting for PID, it is retained in the latents that instead corrected for CNVs

Chapter 4

Evaluating Single-Cell Foundation Models for Representation Learning

Large-scale foundation models, which are pre-trained on massive, unlabeled datasets and subsequently fine-tuned on specific tasks, have recently achieved unparalleled success on a wide array of applications, including in healthcare and biology [10, 13–17]. The success of these models has showcased the power of leveraging generalizable features and contextual understanding to improve a model's performance.

Foundation models for single-cell RNA sequencing data could substantially improve the performance of single-cell RNA-sequencing analysis pipelines, and several such models have recently been developed, such as scBERT [7], scGPT [108], Geneformer [109], scFoundation [110], UCE [9], and scSimilarity [111]. The pre-train then fine-tune paradigm employed by these models holds the promise of allowing practitioners to leverage vast amounts of single-cell RNA sequencing data collected across a variety of technical and biological conditions, captured in pre-trained gene or cell embeddings. Fine-tuning the models on downstream tasks of interest then theoretically requires collection of much smaller, application-specific datasets than would be required if training a model from scratch. This sample efficiency is of particular value in the biological domain, where clinical data or data whose labels are determined experimentally can be challenging or expensive to collect, resulting in smaller available datasets for downstream tasks of interest.

In this chapter, we explore two such models, scBERT [7] and scGPT [108]. Focusing on the fine-tuning task of cell type annotation, we provide further baselines and ablation studies beyond what was studied in the original papers, in order to examine the benefits of these foundation models for this task. We also explore the pre-training paradigm implemented by scBERT, and show that simple heuristics can achieve good performance at the pre-training task as it is currently formulated. Finally, using scBERT as an example, we demonstrate the

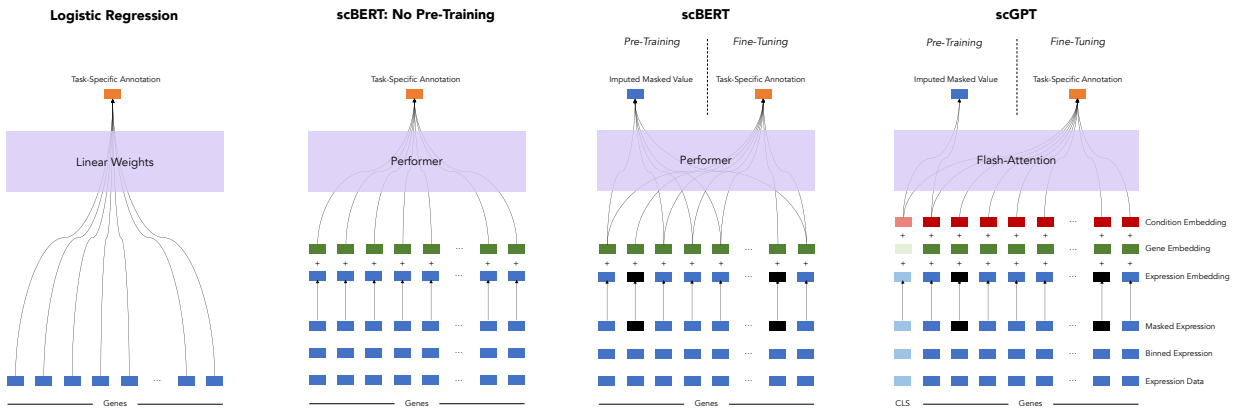


Figure 4.1: Schematic of different methods compared in this work. Two foundation models, scBERT and scGPT, construct embeddings of input data and process them via Transformer-based architectures. The architectures are trained via a masking task and fine-tuned for a task of interest. We compare these models to logistic regression and a Transformer architecture without pre-training to understand the potential benefits of the pre-train/fine-tune paradigm in several different experimental settings.

potential sensitivity of fine-tuning to hyperparameter settings and parameter initializations. Taken together, our results highlight the importance of rigorously testing foundation models against well established baselines, establishing challenging fine-tuning tasks on which to benchmark foundation models, and performing deep introspection into the embeddings learned by the model in order to more effectively harness these models to transform single-cell data analysis.

4.1 Methods

4.1.1 Models Studied

Figure 4.1 overviews the different models studied in this analysis. The two foundation models studied in this work, scBERT and scGPT, rely on Transformer-based architectures [112] for processing embedded representations of the input gene expression data, but differ in how they represent input data, their model architectures, and training procedures. We also analyze a logistic regression baseline that directly trains on a specific task of interest without pre-training.

scBERT. scBERT [7] embeds each gene as the sum of two embeddings: one representing the gene's normalized, binned log-scale expression level, and a second which specifies the gene identity via the previously developed gene2vec [113] embedding, allowing the model to

differentiate genes in the otherwise order-less data. The resulting embeddings are processed via a Transformer architecture that has been optimized for speed and memory, called the Performer [114].

Closely mimicking the pre-training strategy in BERT [10], scBERT is pre-trained via an imputation task on 1.12 million human cells where "masked" gene expression values are predicted as a function of all other gene embeddings in a cell (see Supplementary Methods). During fine-tuning, a 3-layer neural network is trained on smaller, labeled datasets to take the gene embeddings outputted by the Transformer layers and predict cell type. In our analysis, we also evaluate a non-pre-trained version of scBERT, in which pre-training is skipped, and instead the Transformer layers are initialized with random weights prior to fine-tuning.

scGPT. scGPT largely follows a similar architectural and pre-training paradigm to scBERT, with some different design choices. Instead of binning the input data on the log-scale, scGPT bins genes according to their expression such that genes are evenly distributed across each bin. In lieu of gene2vec, scGPT uses a random gene identity embedding and incorporates an additional "condition embedding" describing meta-information to differentiate each gene. Instead of the long-range Performer architecture, scGPT processes the embeddings via Flash-Attention [115] blocks. In addition to gene embeddings, scGPT further trains a CLS token which summarizes each cell. scGPT implements a *generative* masked pre-training, using a causal masking strategy inspired by OpenAI's GPT series [11].

This model is pre-trained on 33 million human cells and evaluated via a wider suite of fine-tuning tasks, including cell type annotation, genetic perturbation prediction, batch correction, and multi-omic integration. We focus on cell type annotation in this analysis.

Logistic Regression. Unlike the foundation models, the logistic regression baseline [116] does not adhere to a pre-train/fine-tune paradigm. This method has the fewest number of trainable parameters and simply estimates the linear coefficients of the log-normalized gene expression data that best predict the labels of interest, in this case cell types. We use L1-regularization [117] to encourage sparsity.

4.1.2 Experiment Roadmap

In this work, we first explore the relative performance of pre-trained models like scGPT or scBERT compared to a simple baseline, L1-regularized logistic regression, for the fine-tuning task of cell type annotation. Since pre-trained representations can be especially useful for few-shot prediction, in which limited training data is available, we additionally evaluated each model's ability to annotate cell types in the few-shot setting. We find that a simple

logistic regression baseline is competitive with the pre-trained models for annotating cell types even in the few-shot setting.

Next, we sought to understand how much pre-training contributed to the performance of the Transformer-based models. To this end, we skipped the pre-training procedure and directly fine-tuned the model for cell type annotation. We show that for scBERT, removing pre-training does not meaningfully affect the model's performance on cell type annotation, while for scGPT, it does. We further demonstrate that scBERT can achieve good accuracy on masked pre-training and cell type annotation without learning meaningful gene representations, illustrating that good accuracy does not necessarily imply rich representation learning.

Finally, we explore scBERT's robustness to hyperparameter settings and random seeds, and we find that these settings significantly affect learning dynamics and model performance.

4.2 Results

4.2.1 Logistic regression outperforms foundation models for the fine-tuning task of cell type annotation in a dataset-dependent manner.

We ran L1-regularized logistic regression [118] as a linear, interpretable baseline for the cell type annotation tasks explored in the scBERT and scGPT papers, predicting cell types from log-transformed, library size-normalized gene expression values.

As a comparison for scBERT, we evaluated the performance of logistic regression for predicting cell types in the Zheng68K peripheral blood mononuclear cells (PBMC) dataset [119], which contains 68,450 labeled cells, and in the MacParland liver dataset, which contains 8,444 labeled cells from human liver tissue (80% of each dataset used for training and 20% for validation). These datasets were used for fine-tuning in the scBERT paper. In the PBMC data, logistic regression outperformed scBERT in terms of both accuracy and class-averaged (macro) F1 score, a metric which reflects both precision and recall and can better assess performance in cases of class imbalance (Table 4.1). Even for difficult-to-distinguish CD8+ cytotoxic T cells and CD8+/CD45RA+ T cells [7], logistic regression outperformed scBERT in terms of accuracy and F1 score (Table 4.1). In the liver dataset, logistic regression performed comparably to scBERT (Table 4.1).

Similarly, we ran logistic regression as a baseline for the cell type annotation task explored in the scGPT paper. The authors of scGPT evaluated the model's cell type annotation capabilities on three different datasets: multiple sclerosis [120], pancreas [121], and myeloid data [122] (additional dataset details provided in Supplementary Methods). We found

Table 4.1: **Logistic regression baseline for the cell type annotation task in scBERT.** Validation set accuracy and macro F1 score for the cell type annotation fine-tuning task on the PBMC and MacParland liver datasets, for three different models: scBERT as reported in the original publication [7], our reproduction of scBERT trained with the same data and model architecture, and our L1-regularized logistic regression baseline. The original scBERT paper reported results from five different runs of fine-tuning; we ran scBERT fine-tuning with 10 different random parameter initializations and shufflings of the data during training; mean ± 95% confidence intervals reported. Logistic regression outperformed scBERT across all metrics for this task.

Dataset	**Model**	**Accuracy** (↑)	**Macro F1** (↑)	**Accuracy** (↑): 'hard to predict'	**Macro F1** (↑): 'hard to predict'
PBMC	scBERT (reported)	0.759 ± 0.010	0.691 ± 0.002	0.801	0.788
	scBERT (reproduced)	0.766 ± 0.012	0.675 ± 0.012	0.765 ± 0.030	0.782 ± 0.013
	L1 logistic regression	**0.811**	**0.707**	**0.848**	**0.828**
Liver	scBERT (reported)	0.976 ± 0.004	0.959 ± 0.006	–	–
	scBERT (reproduced)	0.974 ± 0.016	0.947 ± 0.039	–	–
	L1 logistic regression	**0.985**	**0.976**	–	–

Table 4.2: **Logistic regression baseline for the cell type annotation task in scGPT.** Accuracy and macro F1 scores for the cell type annotation fine-tuning task on the three datasets evaluated in the scGPT paper. Each deep model was run with 10 different random parameter initializations and shufflings of the data during training; mean performance ± 95% confidence intervals across these runs are shown. The best metric for each dataset is bolded.

Dataset name	**Model**	**Accuracy** (↑)	**Macro F1** (↑)
PBMC	logistic regression	**0.811**	**0.707**
	scBERT	0.766 ± 0.012	0.675 ± 0.012
multiple sclerosis	logistic regression	0.812	0.642
	scGPT	**0.859 ± 0.010**	**0.720 ± 0.023**
pancreas	logistic regression	**0.973**	0.718
	scGPT	0.965 ± 0.005	**0.748 ± 0.023**
myeloid	logistic regression	**0.683**	**0.364**
	scGPT	0.627 ± 0.018	0.340 ± 0.008

that for the multiple sclerosis data, scGPT outperformed logistic regression; for the myeloid data, logistic regression outperformed scGPT; and for the pancreas data, the two methods performed similarly (Figure 4.2; Table 4.2).

These results show that classical baseline methods that are simpler to implement and less expensive to train may outperform or perform comparably to foundation models for certain predictive tasks on single-cell data.

4.2.2 Logistic regression can outperform foundation models even in the "few-shot" setting.

A strong foundation model trains data representations that can easily adapt to downstream prediction tasks, even in "few-shot" settings with minimal training data [14]. Thus, while logistic regression outperformed pre-trained models for cell type annotation in some datasets (PBMC, myeloid), we hypothesized that pre-trained models may show superior performance in the few-shot setting, as they can leverage the representations they learned from vast amounts of unlabeled single-cell data, while logistic regression only has access to the small set of training examples.

We found, however, that across models, datasets, and fractions of training data used, the trends from Section 4.2.1 generally held even in the few-shot setting, with a few exceptions. In particular, as in Section 4.2.1, logistic regression outperforms or matches the foundation models on the PBMC, liver and myeloid data even with limited training data (Figures 4.2, 4.3). Similarly, just as in Section 4.2.1, scGPT outperformed logistic regression on the multiple sclerosis data across all training set sizes, with the relative benefit of the pre-trained model increasing with decreasing training set size (Figure 4.2). For the pancreas dataset, logistic regression and scGPT were closely matched with larger training set sizes, but logistic regression outperformed scGPT once the training set was limited to just 10% or 25% of the original training set size (Figure 4.2). Detailed performance metrics and two-tailed t-test p-values describing these comparisons are shown in Figures 4.2 and 4.3.

One possible explanation for the different performance trends across these datasets lies in the different nature of the datasets. Among all the datasets, the pre-trained model specifically excelled with the multiple sclerosis dataset, which was crafted by the scGPT authors to represent an "out of distribution" scenario: the fine-tuning training set consisted of neuronal cells from 9 healthy donors, and the test set consisted of neuronal cells from 12 multiple sclerosis patients. The pre-trained model may be better equipped to handle prediction under dataset shift, having been pre-trained on a vast number of cells from across many different clinical contexts, whereas logistic regression has only been trained on a single clinical context that is no longer present in the test set. In other words, pre-training may serve as a form of regularization that protects the model from overfitting to the training distribution and helps it to generalize.

None of the other datasets studied in this work suffer from as drastic a distribution shift. Thus, in these scenarios, the cost of training a high parameter model with a small amount of training data may outweigh the benefit of pre-training, and thus logistic regression outperforms or performs comparably to scGPT. Analyzing the relative performances of the

pre-trained and logistic regression models across these different data settings suggests that pre-trained models have unique strengths that make them the better choice for some, but not all, single cell predictive tasks.

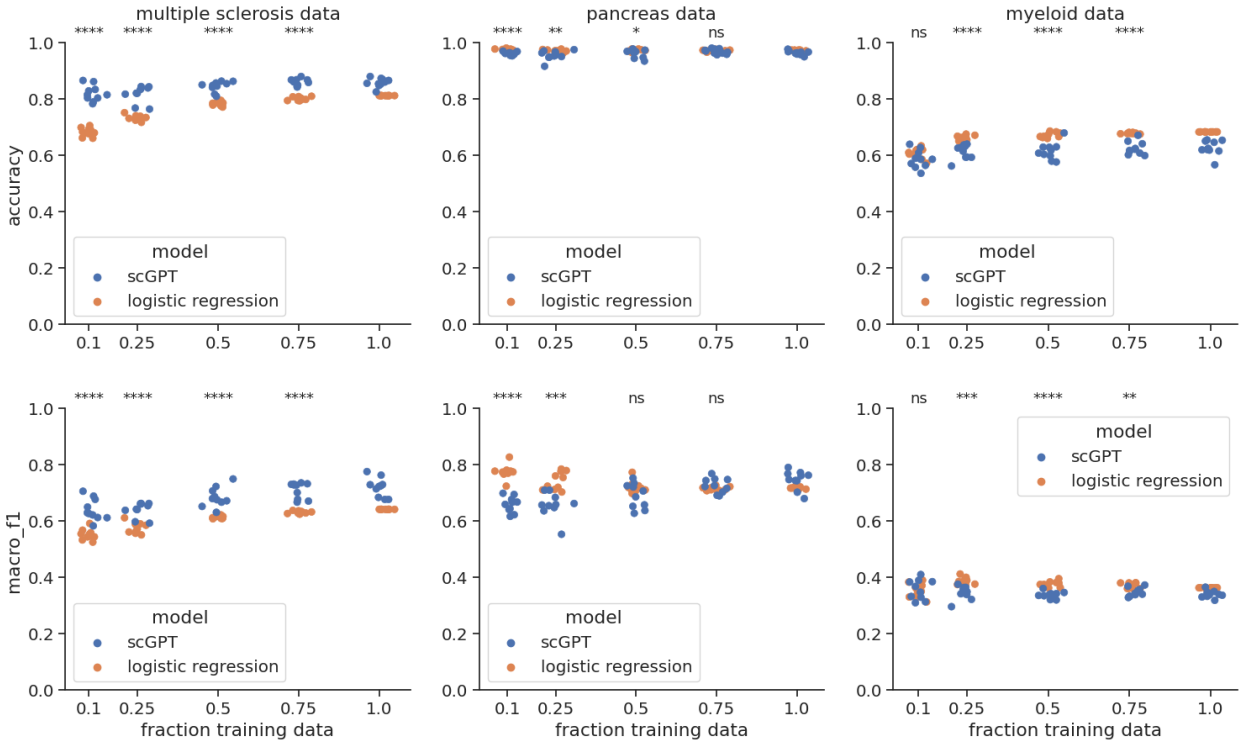


Figure 4.2: scGPT and logistic regression test set performance (accuracy, top; F1 score, bottom) for the fine-tuning task of cell type annotation in the three datasets used in the scGPT paper, using different amounts of training data (jitter added to x-axis to better display overlapping data points). The same pre-trained model was used in all scGPT runs. Each experiment was run 10 times with different random seeds, each time randomly subsetting the training set and, for scGPT, randomly initializing the fine-tuning model parameters. For each dataset, a two-sided t-test was used to compare performance of the models for each amount of training data: **** $p <= 0.0001$; *** $p <= 0.001$; ** $p <= 0.01$; * $p <= 0.05$; "ns" not significant ($p > 0.05$).

4.2.3 Skipping pre-training does not affect fine-tuning performance for scBERT, but does for scGPT.

The paradigm of "pre-train then fine-tune" assumes that the representations learned during unsupervised pre-training help the model perform well on downstream tasks. To test this, we ran an ablation study in which we skipped the pre-training step, effectively using a random embedding of a cell's expression data as input for predicting cell type (see Supplementary Methods).

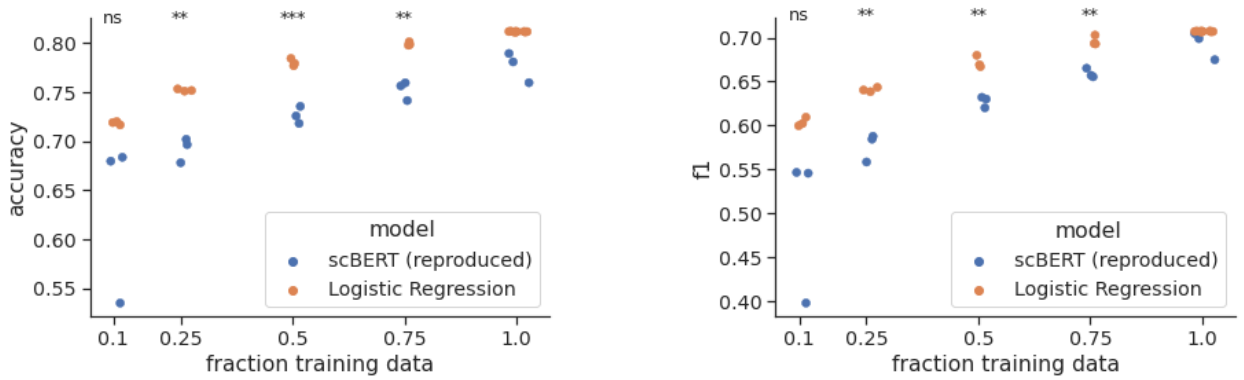


Figure 4.3: scBERT and logistic regression validation set performance (accuracy, left; F1 score, right) for the fine-tuning task of cell type annotation in the PBMC dataset, using different amounts of training data (jitter added to x-axis to better display overlapping data points). The same pre-trained model was used in all scBERT runs. Each experiment was run independently 3 times, each time randomly subsetting the training set and, for scBERT, randomly initializing the fine-tuning model parameters and shuffling the order of the training data. Logistic regression consistently outperformed scBERT even with small amounts of training data. A two-sided t-test was used to compare performance of the models for each amount of training data: *** $p <= 0.001$; ** $p <= 0.01$; "ns" not significant ($p > 0.05$).

Table 4.3: **Ablation studies for the cell type annotation task in scBERT.** Accuracy and macro F1 score for the cell type annotation fine-tuning task on the PBMC and MacParland liver datasets, for three different models: our reproduction of scBERT trained with the same data and model, and ablations that remove the pre-training step and gene2vec embedding, respectively. All results are reported on the validation set. Each model was run with 10 different random parameter initializations and shufflings of the data during training; mean performance ± 95% confidence intervals across these runs are shown. We found that without pre-training, none of the performance metrics significantly changed. Without gene2vec embeddings, accuracy and F1 averaged across all cell types decreased (** indicates p<0.01, *** indicates p<0.001, "ns" indicates "not significant"; two-tailed t-test $p = 3.67 \times 10^{-3}$ and 8.72×10^{-4}, respectively) and F1 for the "hard to predict" cell types decreased (two-tailed t-test $p = 4.10 \times 10^{-3}$).

	Model	**Accuracy** (↑)	**Macro F1** (↑)	**Accuracy (↑):** 'hard to predict'	**Macro F1 (↑):** 'hard to predict'
PBMC	scBERT (reproduced)	.766 ± .012	.675 ± .012	.765 ± .030	.782 ± .013
	No pre-training	.758 ± .014 (ns)	.672 ± .006 (ns)	.754 ± .035 (ns)	.772 ± .016 (ns)
	No gene2vec	.701 ± .040**	.595 ± .042**	.714 ± .046 (ns)	.709 ± .046**
Liver	scBERT (reproduced)	.974 ± .016	.947 ± .039	–	–
	No pre-training	.925 ± .122 (ns)	.875 ± .200 (ns)	–	–

For scBERT, we found that removing pre-training did not significantly change performance for annotating PBMC or liver cell types (Table 4.3; PBMC: two-tailed t-test $p = 0.34$ and $p = 0.59$ for accuracy and F1 score, respectively; liver dataset: two-tailed t-test $p = 0.42$ and $p = 0.46$, respectively). This finding differs from the results reported by the authors in their Extended Data Figure 1a, which we were not able to reproduce, and it held true whether we froze (did not update) most of the transformer weights during fine-tuning or allowed them all to be updated (Table 4.4). Our findings indicate that the success of scBERT on a downstream task like cell type annotation cannot be directly attributed to effective pre-training without further investigation, such as an experiment which ablates pre-training. Even though pre-training did not benefit the downstream tasks probed here, further analysis is needed to understand whether meaningful cell representations were learned during pre-training, as it is possible that meaningful representations were learned but were not necessary for this simple task.

Table 4.4: Here, we annotate our scBERT PBMC results to show that results did not significantly change whether we allowed updates to ("unfroze") one (black), two (blue), or all (orange) transformer layers during fine-tuning. In our main results, we froze all but the final transformer layer during fine-tuning. As the original scBERT authors had frozen all but the final two layers, we also ran a version of the model leaving two layers unfrozen. Additionally, in a correspondence with the scBERT authors, they claimed to have unfrozen all transformer layers during their "no pretraining" experiment, so we added this variant to our "no pretraining" experiments (in orange). As in Table 4.1 and Table 4.3, we compared performance metrics from the "no pre-training" and "no gene2vec" experiments to the corresponding metrics from "scBERT (reproduced)" and we found that whether one, two, or all transformer layers were unfrozen during fine-tuning, our main results were unchanged: without pre-training, none of the performance metrics significantly changed; without gene2vec embeddings, accuracy and F1 averaged across all cell types decreased (** indicates p<0.01; *** indicates p<0.001; "ns" indicates "not significant"). Importantly, we always compared models with the same numbers of frozen weights (i.e. blue "no pre-training" vs. blue "scBERT (reproduced)"; orange "no pre-training" vs. orange "scBERT (reproduced)") to control for the effect of the number of trainable parameters on performance.

Model	**Accuracy**	**Macro F1 score**	**Accuracy** "hard to predict" cells	**Macro F1 score** "hard to predict" cells
scBERT (reported)	0.759 ± 0.010	0.691 ± 0.002	0.801	0.788
scBERT (reproduced)	0.766 ± 0.012	0.675 ± 0.012	0.765 ± 0.030	0.782 ± 0.013
Unfreeze 2 layers	0.764 ± 0.019	0.669 ± 0.021	0.779 ± 0.028	0.781 ± 0.020
Unfreeze all layers	0.784 ± 0.013	0.696 ± 0.013	0.781 ± 0.031	0.799 ± 0.013
L1 logistic regression	**0.811**	**0.707**	**0.848**	**0.828**
No pre-training (random embedding)	0.758 ± 0.014 (ns)	0.672 ± 0.006 (ns)	0.754 ± 0.035 (ns)	0.772 ± 0.016 (ns)
Unfreeze 2 layers	0.764 ± 0.011 (ns)	0.672 ± 0.009 (ns)	0.759 ± 0.025 (ns)	0.780 ± 0.011 (ns)
Unfreeze all layers	0.781 ± 0.011 (ns)	0.694 ± 0.007 (ns)	0.781 ± 0.026 (ns)	0.798 ± 0.012 (ns)
No gene2vec	0.701 ± 0.040**	0.595 ± 0.042***	0.714 ± 0.046 (ns)	0.709 ± 0.046**
Unfreeze 2 layers	0.713 ± 0.028**	0.596 ± 0.047**	0.726 ± 0.026**	0.723 ± 0.034**

By contrast, the scGPT authors report results from a similar experiment, "scGPT (from-scratch)," in their Supplementary Table S1, finding that their pre-trained model outperformed similar models fine-tuned "from-scratch," without pre-training. We independently ran our pre-training ablation on scGPT, and also concluded that pre-training improved performance on cell type annotation for scGPT across all three datasets (Table 4.5 and Figure 4.4). We note that while we reach the same conclusion as Cui et al. [108], our results significantly differ from what was reported in their Supplementary Table S1, which is explained by the fact that the scGPT authors used a smaller model in their no pre-training ablation to optimize performance and create a more competitive baseline.

Table 4.5: **Cell type annotation results when skipping pre-training for scGPT.** Accuracy and macro F1 scores for the cell type annotation fine-tuning task on the three datasets evaluated in the scGPT paper for three different models: scGPT (with pre-training), and two different versions of an ablation skipping pre-training, one in which all weights of the model were learnable during fine-tuning, and one in which all pre-decoder weights were frozen during fine-tuning. Each model was run with 10 different random parameter initializations and shufflings of the data during training; mean performance ± 95% confidence intervals across these runs are shown. The best metric for each dataset is bolded. We note that while we reach the same conclusion as Cui et al. [108] that pre-training benefits downstream performance for cell type annotation, these results significantly differ from what was reported in their Supplementary Table S1; further exploration is needed to understand the source of this discrepancy.

Dataset	Model	Accuracy	Macro F1 score
multiple sclerosis	scGPT	**0.859 ± 0.01**	**0.72 ± 0.023**
	scGPT (no pre-training)	0.074 ± 0.063	0.023 ± 0.028
	scGPT (no pre-training; freeze weights)	0.461 ± 0.176	0.298 ± 0.134
myeloid	scGPT	**0.627 ± 0.018**	**0.34 ± 0.008**
	scGPT (no pre-training)	0.163 ± 0.01	0.025 ± 0.001
	scGPT (no pre-training; freeze weights)	0.172 ± 0.03	0.034 ± 0.02
pancreas	scGPT	**0.965 ± 0.005**	**0.748 ± 0.023**
	scGPT (no pre-training)	0.949 ± 0.011	0.648 ± 0.042
	scGPT (no pre-training; freeze weights)	0.949 ± 0.008	0.627 ± 0.025

4.2.4 For scBERT's embedding scheme and pre-training objective, good pre-training and fine-tuning accuracy can be achieved without learning rich representations

Despite good performance on the pre-training objective (78% accuracy), Section 4.2.3 shows that pre-training scBERT does not affect the final fine-tuning performance. In this section, we

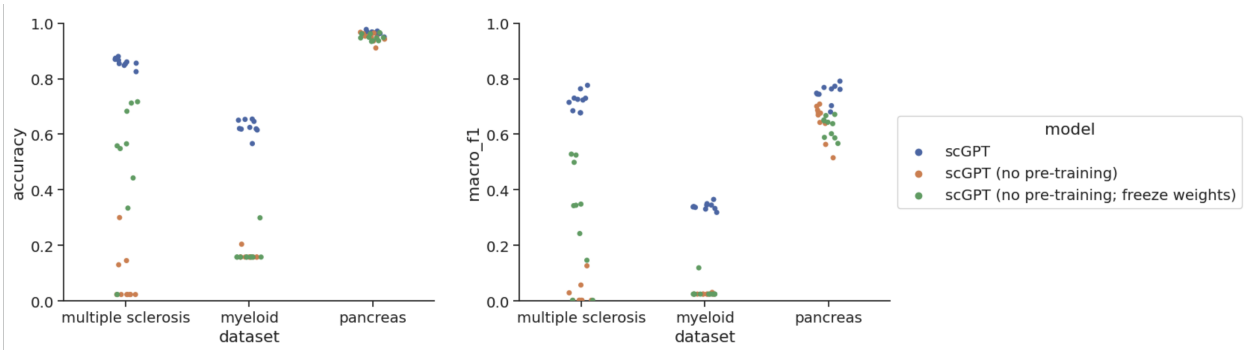


Figure 4.4: Accuracy (left) and macro F1 score (right) for cell type annotation on 3 datasets for the no pre-training ablation in scGPT (with and without freezing weights) vs. vanilla scGPT fine-tuning. Each model was run with 10 different random seed initializations. Mean and 95% confidence intervals for each model are shown in Supplementary Table 4.5. We note that while we reach the same conclusion as Cui et al. [108] that pre-training benefits downstream performance for cell type annotation, these results significantly differ from what was reported in their Supplementary Table S1, which is explained by the fact that the scGPT authors used a smaller model in their no pre-training ablation to optimize performance and create a more competitive baseline.

ablate another component of the scBERT architecture and show that good pre-training or fine-tuning accuracy is not sufficient evidence that a model is learning rich gene representations.

In scBERT, a given gene's embedding is the sum of two separate embeddings for that gene: a "gene2vec" embedding and an "expression" embedding. The "gene2vec" embedding codifies the identity of each gene and is constant for each gene across every cell in the dataset. The "expression" embedding transforms a given gene's expression into one of six randomly-initialized vectors based on its binned expression (the bins are [0-1), [1-3), [3-7), [7-15), [15-31), and 31+ library size-normalized UMIs). The expression embedding—not the gene2vec embedding—drives the variability in the learned representations of a given gene for different cells and is the focus of the masking task used to pre-train the model.

In this experiment, we remove the gene2vec embedding from the representation of each gene. This severely limits the representational capacity of the model, as there are now only six embeddings that are used to represent every gene in the dataset. During pre-training, the model has no knowledge of a gene's identity and every masked gene has identical "context" (note that since Transformers are permutation invariant [112], the model cannot memorize the input positions of genes in order to identify them). Every gene that falls in the same expression bin is represented identically – for example, across the Panglao pre-training dataset [123], 95% of genes are identically represented as falling in the [0,1) bin, 2% of genes in the [1-3) bin, 1.5% of genes in the [3-7) bin, and <1% of genes in the remaining bins.

Given that the model does not know which gene is which, one might expect that scBERT

would not be able to predict a gene's masked expression with high accuracy. We observed, however, that even in this challenging setting, scBERT achieved 72% accuracy for predicting masked gene expression on the Panglao validation set (compared to 78% when genes are represented using both gene2vec + expression embeddings). Upon introspection into the "no gene2vec" experiment, we found that for 99.8% of the 67,880 cells in the Panglao pre-training validation set, scBERT predicted the most common expression bin per cell for all masked genes in that cell (Table 4.6). These results demonstrate that with scBERT's binned expression embedding scheme, a model can rely on simple heuristics such as predicting the most common expression bin to perform well at masked pre-training without learning contextual gene representations.

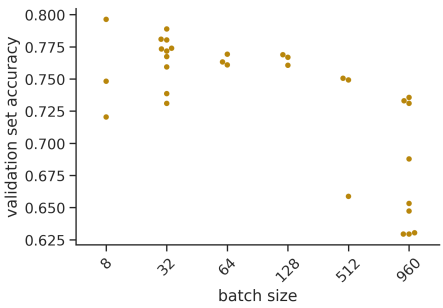
Table 4.6: Validation set accuracy on masked pre-training when gene2vec embeddings are included ("full scBERT") or are not included ("no gene2vec") as part of each gene's input embedding. In the no gene2vec setup, the same expression bin is necessarily predicted for all masked genes in a given cell, as they all have identical context. This is not the case in the full scBERT embedding scheme; hence "full scBERT" has N/A in the column quantifying the fraction of cells for which the most common expression bin is predicted, as only gene-level predictions can be summarized in this way.

Model	**Data**	Pre-training masking accuracy (↑)	Fraction of cells in which most common expression bin was predicted for all genes	of masked genes for which the most common bin (per cell) was predicted
no gene2vec	PBMC	68%	100%	100%
full scBERT	PBMC	79%	N/A	79%
no gene2vec	Panglao	72%	99.8%	99.9%
full scBERT	Panglao	78%	N/A	84%

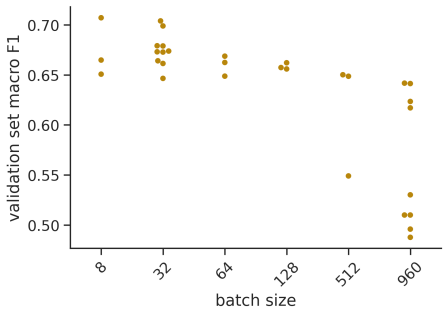
Regarding fine-tuning, scBERT trained without gene2vec embeddings exhibited decreased but still reasonable performance on the cell type annotation task (Table 4.3, "no gene2vec" accuracy 0.701 and F1 0.595). We note that we did not re-tune hyperparameters specifically for the no gene2vec experiment, but rather used the best hyperparameters we identified for fine-tuning the full scBERT model, so it is possible that the performance we report here is an underestimate of the potential performance had we tuned hyperparameters for this task specifically (see Section 4.2.5). While the logistic regression baseline and pre-training ablation already demonstrated that rich representations are not needed for accurate cell type annotation in the PBMC data, this result further reinforces that performance on cell type annotation is not a proxy for the representation learning capabilities of a foundation model.

4.2.5 Robustness to hyperparameter choices and parameter initialization

In working to reproduce scBERT's reported accuracy for cell type annotation in the PBMC data, we found that batch size and learning rate - neither of which were reported in the scBERT paper - were critical hyperparameters to tune in order to achieve optimal fine-tuning results. Adopting a learning rate of 1×10^{-4}, as was the default setting in the scBERT code base, we evaluated validation set performance across multiple batch size settings, and observed that smaller batch sizes, such as 8 or 32, generally achieved better performance (Figure 4.5). We chose to use a batch size of 32 in all our experiments, to strike a balance between efficiency (large batch sizes are more efficient) and optimization (smaller batch sizes achieve more optimal results).



(a) Validation set accuracy for different batch sizes.



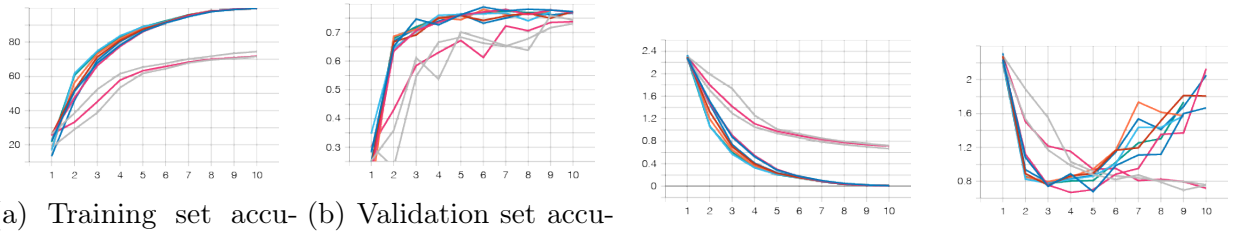
(b) Validation set macro F1 score for different batch sizes.

Figure 4.5: The mini-batch size used for fine-tuning scBERT for the cell type annotation task has a large effect on predictive performance, in terms of validation set accuracy (a) and macro F1 score (b). All models shown here were trained with a learning rate of 1×10^{-4}. Each model was trained for 10 epochs, and metrics from the epoch with the best validation set accuracy are shown. Each hyperparameter setting was run multiple times with different random seeds, each represented by a single dot.

In addition to batch size, we explored the effect of random seed (i.e. parameter initialization) on predictive performance. The random seed determines the initial random parameterization of the fine-tuning components of the scBERT architecture (these include a convolutional filter to summarize each gene embedding, and a 3-layer neural network). Similar to results reported by Liu et al. [124], we found that performance indeed varied across random seeds, and that fluctuations in performance across random seeds were greater for models trained with larger batch sizes (Figure 4.5). Even for an optimal batch size of 32, we observed that two different learning patterns emerged across the random seeds: one group of random seeds led to convergence in fewer epochs and achieved overall higher accuracy after

ten training epochs, while a second group of random seeds led to longer convergence times and overall lower accuracy after ten training epochs (Figure 4.6). Understanding this bimodal behavior of the learning dynamics requires further exploration. We note that if the models with low-performing initializations were trained for additional epochs, they may eventually reach the quality of the high-performing ones.

These findings emphasize that in order for a model to be reproducible, 1) a full set of hyperparameter settings must be transparently reported and 2) performance should be reported over multiple random seeds, as we did in Table 4.1 and Supplementary Table 4.2. While releasing code is incredibly helpful and important, it is not sufficient to only specify hyperparameters as default settings in a code base, as it is all too easy for users to inadvertently change hyperparameter settings when running code in a new compute environment (eg. training with the same batch size setting on 1 GPU vs. 4 GPUs in a distributed fashion may result in batch size effectively being scaled by 4).



(a) Training set accuracy (b) Validation set accuracy (c) Training set loss (d) Validation set loss

Figure 4.6: Accuracy and loss per epoch for fine-tuning scBERT for cell type annotation, for ten different models initialized with different random seeds, but otherwise identical. All models were trained with batch size 32 and learning rate 1×10^{-4} for 10 epochs. We observe that parameter initialization affects both learning dynamics and final performance metrics.

4.3 Discussion

As interest in foundation models for single-cell RNA sequencing data grows within the genomics community [108–111, 124], we aimed to provide a deeper understanding of their potential benefits and limitations by providing additional baselines and introspection into the models' components.

We demonstrated that simple models like logistic regression can be competitive for the fine-tuning task of cell type annotation, even in the few-shot setting. We further showed that even without pre-training, Transformer architectures can perform well at cell type annotation. These results underscore that cell type annotation is not a challenging enough task on which to demonstrate the potential value of single-cell foundation models. The

value of foundation models for single-cell RNA sequencing data will instead be showcased on more challenging fine-tuning tasks for which simple models do not excel, such as annotating cell types under distribution shift or predicting cellular responses to perturbations, or else through the models' abilities to capture and elucidate meaningful gene-gene interactions through their learned attention weights. When claiming the latter, it is important to keep in mind that strong downstream performance does not necessarily imply rich representation learning, as shown in our "no pre-training" and "no gene2vec" ablation experiments. Finally, since hyperparameter settings and the initial parameterization of models weights can have large effects on the learned model, it is important to have transparent reporting of all design choices and hyperparameter settings, in addition to sharing code and trained models, in order for the community to reproduce and build off of each other's works.

Taken together, we hope that these results shed light on the importance of deep introspection into model performance, as well as rigorous baselines and evaluation tasks, to demonstrate the value and justify the compute costs of large scale foundation models for RNA sequencing data. We hope that these findings will enable and encourage the community to improve on current Transformer-style architectures for single-cell RNA sequencing data.

4.4 Supplementary Methods

4.4.1 Additional dataset details

Multiple Sclerosis. The multiple sclerosis data was comprised of neuronal cell types from healthy donors and patients with multiple sclerosis. We downloaded the pre-processed data provided by the scGPT authors, thus using the same pre-processing and train/test splits as they did. The train/test split for this dataset simulates dataset shift, with cells from 9 healthy donors comprising the training set, and from 12 multiple sclerosis patients comprising the test set. There are 7,844 cell in the training set and 13,468 in the test set. The ground truth cell-type labels were taken from the original publication of the data [120].

Myeloid. This dataset contains tumor infiltrating myeloid cells from multiple different tumor types [122]. We downloaded the pre-processed data provided by the scGPT authors, thus using the same pre-processing and train/test splits as they did. In particular, the dataset consisted of cells from nine different cancer types, with the train set containing cells from cancer types UCEC, PAAD, THCA, LYM, cDC2, and KIDNEY, and the test set containing cells from MYE, OV-FTC, and ESCA. The final cell counts are 9,748 in the training set and 3,430 in the test set.

Pancreas. The pancreas dataset contained human pancreas cells from five different scRNA-seq studies, which were reprocessed by Chen at al. [121]. The training set consisted of 10,600 cells of 13 cell types (alpha, beta, ductal, acinar, delta, PSC, PP, endothelial, macrophage, mast, epsilon, schwann, and T cell) from two datasets, and the test set consisted of 4,218 cells of 11 cell types (alpha, beta, ductal, PP, acinar, delta, PSC, endothelial, epsilon, mast, MHC class II) from the other three datasets. We downloaded the pre-processed data provided by the scGPT authors, thus using the same pre-processing and train/test splits as they did.

4.4.2 Pre-training and fine-tuning scBERT

In the original scBERT paper, the scBERT model was pre-trained on the Panglao dataset provided by the scBERT authors, which is comprised of gene expression from multiple studies spanning 74 human tissues. The model was then fine-tuned on individual smaller datasets, for example the Zheng68k PBMC dataset, and metrics were reported for the performance of the model in annotating cell types in the fine-tune dataset. In our experiments, we ran pre-training on the Panglao dataset (provided by the original scBERT authors upon request, with 95% of the data used for training and 5% for validation). We masked the expression embedding for 15% of genes in a given cell (i.e. set the embedding to a randomly-initialized "mask" embedding). Genes with log-normalized UMIs between [0,1) were excluded from the masking task in scBERT in order to avoid an overwhelming class imbalance, since >95% of the expression values in the Panglao dataset were zero. We pre-trained the model using the Adam optimizer [125] in PyTorch, with a batch size of 8, gradient accumulation every 60 steps and a learning rate of 1×10^{-4} for 17 epochs.

After pre-training the model, we ran fine-tuning on the Zheng68K PBMC dataset (80% training, 20% validation), using the scripts provided by the authors, with minimal changes. For fine-tuning, we fixed the training and validation sets to be constant across different runs and models, so that our evaluations would isolate differences in performances due to changes in the model, rather than changes in the training or validation data. We trained all fine-tuning experiments using the Adam optimizer [125] in PyTorch with a learning rate of 1×10^{-4} and batch size of 32 (with no gradient accumulation) for 10 training epochs, and reported results from the epoch corresponding to the best validation set accuracy.

During fine-tuning, we 'froze' (i.e. did not apply gradient updates to) all transformer weights except for those in the final transformer layer, to reduce the number of trainable parameters in the model; we also experimented with freezing all weights except for those in the final two layers, and with unfreezing all weights, and found no significant difference in performance. With all transformer weights unfrozen, we found that for about 20% of the

random seeds we tested, the model failed to train effectively, with very low (approx. 30%) accuracy for both the train and validation sets, and we removed these runs from our analysis as outliers. We note that we had simply re-used the best hyperparameters from our main fine-tuning experiments, where most transformer weights were frozen, but it is likely that these failing runs could be avoided by tuning hyperparameters for this setting specifically.

For pre-training and fine-tuning, we slightly modified the scBERT authors' provided code to use pyTorch's DistributedDataParallel module (https://pytorch.org/tutorials/intermediate/ddp_tutorial.html) to allow us to train over multiple GPUs in our compute environment. For fine-tuning, we further adjusted the code to weight the loss for different classes differently in order to counter class imbalance and improve the model's ability to generalize outside the training data, and we modified the dataloader to load the same training samples in each epoch of training (shuffling their order), whereas the original scBERT code used a different sampling of the training data in each epoch.

4.4.3 Fine-tuning scGPT

We followed the fine-tuning procedure outlined in the original scGPT paper and implemented in the tutorial provided by the authors at https://github.com/bowang-lab/scGPT/blob/main/tutorials/Tutorial_Annotation.ipynb. For few-shot experiments, we randomly subsampled the training datasets in a stratified fashion to ensure that the same class balance was present in the full and subsampled training data. We used all default settings and hyperparameters provided in the authors' code, including a learning rate of 1×10^{-4} and batch size of 32. For cell type annotation in the full-data and few-shot settings, we fine-tuned for 20 training epochs and selected the model from the epoch with the best validation set performance for reporting test set results. For fine-tuning the "no pre-training" experiment, we fine-tuned for 30 training epochs, in case further training was needed to compensate for the lack of pre-trained embeddings.

4.4.4 Logistic Regression

For our logistic regression baseline, we split the data into an 80:20 train:test split and performed 5-fold cross validation using the training data to choose the regularization coefficient. For experiments that used 100% of the training data, the following regularization coefficients ("c" in sklearn's sklearn.linear_model.LogisticRegression function) were chosen:

- PBMC: c=0.1
- multiple sclerosis: c=0.1

- pancreas: c=100
- myeloid: c=0.1

For the few-shot experiments, a different c was chosen using cross validation for each subsample of the training data.

4.4.5 Few-shot experiments

To evaluate performance with limited training data, we trained each model on 75%, 50%, 25%, and 10% of its full training data. We used the same train/test split as in our other fine-tuning experiments, randomly sub-sampling a fraction of the original training data. For each new subsample of training data, we performed 5-fold cross validation using the smaller training data to choose the regularization coefficient. We evaluated performance on the full test set. We ran each experiment 3 times for the PBMC data and 10 times for the multiple sclerosis, pancreas, and myeloid datasets, each time using a different random sub-sample of the training data and a different random seed to initialize parameters, to capture the variation resulting from both the composition of the small training set and the model's initial parameter settings.

4.4.6 "No pre-training" ablation

For the "no pre-training" ablation experiment on scBERT, we randomly initialized the full scBERT architecture, but instead of pre-training, we directly fine-tuned this model on the PBMC data for the cell type annotation task. Similar to our other runs of pre-training, we froze most of the (random, in this case) weights of the transformer, leaving only the weights in the last layer of the transformer and in the fine-tuning portions of the architecture to be updated during fine-tuning. We fine-tuned each model for 10 epochs, and reported results for the epoch with the highest validation set accuracy.

For the "no pre-training" ablation experiment on scGPT, we took a very similar approach, randomly initializing the full scGPT architecture and directly fine-tuning the model for cell type annotation on either the myeloid, multiple sclerosis, or pancreas dataset without learning or loading any pre-trained weights onto the model. We experimented with two different modes of this experiment, one in which all weights in the model were trained during fine-tuning, and one in which all pre-decoder weights were frozen, reducing the number of trainable parameters from 51,342,358 to 19,999,766. We fine-tuned each model for 30 epochs, and reported results for the epoch with the highest validation set accuracy.

4.5 Code Availability

Software and analysis code is publicly available at https://github.com/clinicalml/sc-foundation-eval.

Chapter 5

Learning Patient-Level Representations for Clinical Prediction

Single-cell RNA sequencing (scRNA-seq) offers insights into cellular heterogeneity and tissue composition, yet leveraging this data for patient-level clinical predictions remains challenging due to the set-structured nature of single-cell data, as well as the scarcity of labeled samples. To address these challenges, we introduce scSet, a novel diffusion-based autoencoder that learns patient-level representations from sets of single-cell transcriptomes. Our method uses a transformer-based encoder to process variably sized and unordered cell inputs, coupled with a conditional diffusion decoder for self-supervised learning on unlabeled data. By pre-training on large-scale unlabeled datasets, scSet generates robust patient representations that can be fine-tuned for downstream clinical prediction tasks. We demonstrate the effectiveness of scSet patient embeddings for clinical prediction across multiple real-world datasets, where they outperform existing patient representations, even with limited labeled data. This work represents an important step toward bridging the gap between single-cell resolution and patient-level insights.

5.1 Introduction

Single-cell RNA sequencing (scRNA-seq) provides a detailed view of the cellular composition of a tissue, enabling insights such as the identification of cell type specific biomarkers [30, 126], tumor heterogeneity [30, 127], the composition of the tumor-immune microenvironment [60, 128], and the diverse cell states that can exist for a single-cell type [129, 130]. While it is widely acknowledged that single-cell features correlate with clinical outcomes of interest [131], few ML tools exist to make patient-level predictions based on scRNA-seq data. We identify two major hurdles for ML for patient-level prediction from single-cell data: (1) Patient single-cell

datasets do not immediately lend themselves to predictive machine learning methods which assume that the input features are either consistently or meaningfully ordered (e.g. logistic regression or recurrent/convolutional networks, respectively), since their features (the cells) have no consistent or meaningful order. (2) The numbers of single-cell samples with clinical labels for any given disease state is often too small to employ machine learning methods to automate the discovery of features which correlate with clinical outcomes of interest. This is due to a combination of single-cell data being expensive to generate, clinical samples requiring patient consent and potentially invasive biopsies, and clinical labels requiring careful annotation and patient follow-up.

Most commonly, single-cell samples are simply averaged across cells prior to being input to an ML model, or else manual feature engineering and statistical analysis are used to find correlations between single-cell features with patient-level information [132, 133]. Machine learning architectures that predict sample-level information from a parametrized embedding of single-cell data have only recently started to emerge [134–136].

In this work, we introduce scSet, a diffusion-based autoencoder for learning patient-level representations from scRNA-seq data. Our method addresses both of the above challenges, first by developing an encoder that can handle variably sized and unordered cell inputs, and second by leveraging unlabeled samples for self-supervised learning via a denoising diffusion objective. Taken together, scSet learns patient representations in an unsupervised manner, which can then be fine-tuned to predict clinical features of interest from a limited cohort of clinically-labeled samples.

5.2 Related Work

5.2.1 Representation Learning for Sets

Representation learning for sets has been explored through various approaches, both supervised and unsupervised, with applications spanning multiple domains, including point clouds, graphs, and multi-instance learning (MIL). Early work such as DeepSets [137] proposed permutation-invariant architectures that aggregate unordered inputs using pooling functions like sum or mean. Transformer-based models for sets, such as Set Transformer [138] and Attention-based Deep Multiple Instance Learning [139], introduced attention mechanisms to capture higher-order interactions between set elements. More recently, unsupervised approaches such as SetVAE [140] incorporate principles from Set Transformer into a variational autoencoder framework to learn unsupervised latent representations of sets, using the Chamfer Distance as a proxy reconstruction loss to handle unordered set data. The Chamfer Distance is a

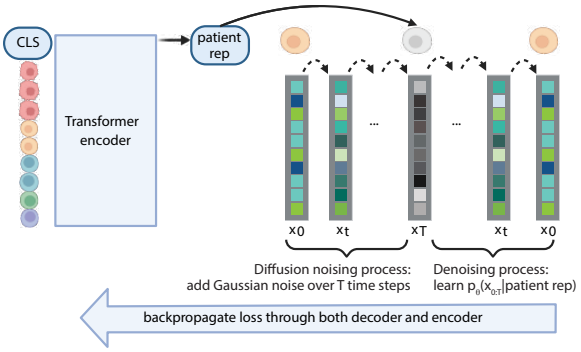
permutation-invariant metric that computes the average squared distance between each point in one set and its nearest neighbor in the other set, making it well-suited for comparing point clouds or sets of embeddings [141]. A noise prediction loss as used in diffusion models [142, 143] is less computationally expensive and can also handle unordered set data, and thus we were motivated to explore a diffusion-based decoder for unsupervised set learning. Others have recently begun to explore this direction too, with applications in 3D point clouds instead of biology [144]. Our work builds on these foundational principles but extends them to the biomedical domain, enabling unsupervised patient representation learning from sets of single cells.

5.2.2 Representation Learning for scRNA-seq

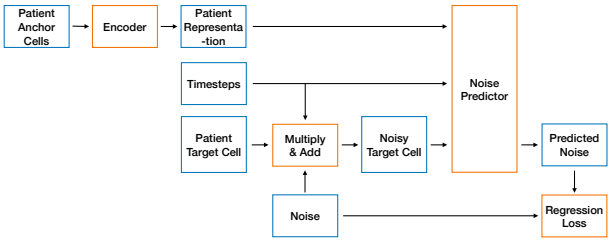
Representation learning in the single-cell space has mostly focused on learning representations of cells, with methods such as scVI [12], Geneformer [145], and scGPT [8], as well as multimodal models such as totalVI [146]. Any of these cell embeddings can be used as input to our model, which instead focuses on encoding a set of cells into a patient-level representation, and decoding a patient representation back to individual cells through conditional denoising diffusion. Other approaches for learning sample-level encodings of single-cells have only recently started to emerge and have focused on supervised methods, as described in Section 5.2.3.

5.2.3 Patient-Level Representations from scRNA-seq

The simplest and most common method for summarizing sample expression is to take the average gene expression across all cells in the sample, referred to as pseudobulk. However, pseudobulk obscures the granular view of cell states afforded by single-cell. Recently, a few methods have been proposed for learning patient-level representations from scRNA-seq data [134–136, 147–150]. These have mostly built off of the deep set [137] or attention-based multiple instance learning (ABMIL) [139] frameworks. While these works propose architectures that learn to aggregate single-cell data into patient-level representations, they are all trained on (semi-)supervised tasks. By contrast, a key contribution of our work is our self-supervised training objective, which allows learning representations from unlabeled data and improves the quality of our downstream supervised predictions.



(a) A set of cells are embedded to a patient representation using a transformer. The patient representation conditions a diffusion model that learn to denoise individual cells, effectively learning $p_\theta(x_{0:T}|\text{patient rep})$.



(b) Model and training components are shown in orange, inputs and outputs in blue.

Figure 5.1: Two complementary overviews of scSet model.

5.2.4 Diffusion Models for scRNA-seq data

Diffusion models have been used for a variety of tasks in machine learning, ranging from image generation [142] to drug discovery [151]. Recently, diffusion models have been applied to scRNA-seq data for gene expression imputation [152–154]. While these methods use diffusion models to generate scRNA-seq data, they do not condition the model on a patient-specific representation, or leverage it as part of an autoencoding framework for learning a patient representation.

5.3 Method

Single-cell RNA sequencing profiles the transcriptomes of individual cells in a patient sample. Each cell's transcriptome is represented as a vector of gene expression values, x, of length G, the total number of all genes detected across cells in our dataset. A scRNA-seq sample from a given patient is observed as an unordered set of single-cell transcriptomes, $X = \{x_i\}_{i=1}^{N}$. Our goal is to learn a meaningful vector representation z of the set of cells for each patient.

To this end, we propose scSet, a diffusion-based autoencoding framework for learning patient-level representations from scRNA-seq data. The following sections detail the decoder, encoder, and training procedure for scSet. A schematic overview of the model is provided in Figure 5.1.

5.3.1 Diffusion-based Decoder

We employ a conditional Denoising Diffusion Probabilistic Model (DDPM) [142, 143], which uses the patient representation z for conditioning and, starting from noise, generates sample cells matching the patient profile.

Given a number of time steps $T \in \mathbb{N}$ and a variance schedule $\beta_1, ..., \beta_T \in \mathbb{R}_{>0}$, we model the diffusion forward-process as

$$q(x_{0:T}|z) = q(x_0|z) \prod_{t=0}^{T-1} q(x_{t+1}|x_t, z) \tag{5.1}$$

where $q(x_0|z)$ is the data distribution of cell profiles conditioned on a patient representation z and

$$q(x_{t+1}|x_t, z) = \mathcal{N}(x_{t+1}; \sqrt{1-\beta_{t+1}}x_t, \beta_{t+1}I). \tag{5.2}$$

As a decoder, we learn a denoising backward-process

$$p_\theta(x_{0:T}|z) = p_\theta(x_T|z) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t, z) \tag{5.3}$$

parametrized as

$$p_\theta(x_T|z) = \mathcal{N}(x_T; 0, \sigma^2 I) \tag{5.4}$$

$$p_\theta(x_{t-1}|x_t, z) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t, z), \sigma_t^2 I), \tag{5.5}$$

with mean predictor $\mu_\theta(x_t, t, z)$ given by

$$\mu_\theta(x_t, t, z) = \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(x_t, t, z) \right). \tag{5.6}$$

Here $\epsilon_\theta(x_t, t)$ is a noise-predicting neural network, $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$.

A key assumption of our paper is that cells in a sample are conditionally independent given z, and thus the probability of the set of cells X is the product of the probability of each cell in the set, $p_\theta(X|z) = \prod_{x \in X} p_\theta(x_t = x|z)$. Due the presence of multiple cell types in each sample, we expect $p_\theta(x_t = x|z)$ to be a complex, multimodal distribution.

The noise-predicting network is an adapted multilayer perceptron with residual connections. The patient representations and sinusoidal time step embeddings are each processed by feed-forward networks, summed and incorporated into the noise prediction through Adaptive

Layer Normalization [155]. For the diffusion process, we use a cosine noise schedule [143] and $T = 1000$ time steps.

5.3.2 Transformer-based Encoder

The scSet encoder, f_ϕ, maps the unordered set of cells in a patient sample $\{x_i\}_{i=1}^N$ to a fixed-dimensional representation $z \in \mathbb{R}^d$, where $d = 256$ in our model. To address the challenges posed by the variable size and lack of ordering in the input data, we employ a transformer-based architecture with a learnable `[CLS]` token that serves as a global representation of the input set.

The architecture begins with a linear embedding layer that projects each cell $x_i \in \mathbb{R}^G$ into d-dimensional space:

$$x'_i = \text{Linear}(x_i), \quad x'_i \in \mathbb{R}^d.$$

A learnable `[CLS]` token is appended to the set of cells, forming the input to the encoder:

$$X' = [\texttt{[CLS]}; x'_1; x'_2; \ldots; x'_N] \in \mathbb{R}^{(N+1)\times d}.$$

The transformed set X' is then passed through a series of transformer encoder blocks [156], each consisting of multi-head self-attention, feedforward networks, and layer normalization. These layers are intended to model interactions between cells and to encode information about cells in the context of their tissue environment. Formally, each encoder layer is defined as:

$$X'^{(\ell+\frac{1}{2})} = \text{LN}(\text{MHSA}(X'^{(\ell)}) + X'^{(\ell)}) \tag{5.7}$$

$$X'^{(\ell+1)} = \text{LN}(\text{FFN}(X'^{(\ell+\frac{1}{2})}) + X'^{(\ell+\frac{1}{2})}) \tag{5.8}$$

where MHSA denotes multi-head self-attention, LN Layer Normalization and FFN a feed-forward network. Dropout is applied to attention and feed-forward layers to prevent overfitting.

After passing through L transformer blocks, the embedded `[CLS]` token is extracted as the patient representation z. This representation z serves as the input to the diffusion-based decoder for patient-specific cell generation, and the encoder is jointly optimized with the decoder during training, as described in Section 5.3.3.

When using z as input for a downstream clinical prediction task, the weights of the encoder can optionally be further updated to tailor the patient embedding for the downstream task (see Section 5.4.2).

5.3.3 Training Procedure

The encoder f_ϕ and the decoder p_θ are jointly trained using the noise prediction loss

$$L(\phi, \theta) = \mathbb{E}_{t\sim[1,T],x_0,\epsilon_t}\left[\|\epsilon - \epsilon_\theta(x_t, t, f_\phi(\{\tilde{x}_i\}_{i=1}^{N_{\text{anchor}}}))\|^2\right] \tag{5.9}$$

where $t \sim \mathcal{U}(\{1, ..., T\})$, $\epsilon \sim \mathcal{N}(0, I)$, $x_t \sim q(x_t|x_0, z)$, and x_0 as well as $\tilde{x}_i$ for $i \in 1, ..., N_{\text{anchor}}$ are drawn uniformly without replacement from the cells of patient j. To estimate the loss during training in practice, mutually exclusive subsets of X are used as input to the encoder ($\{\tilde{x}_i\}_{i=1}^{N_{\text{anchor}}}$, referred to as "anchor cells") and to the noise-prediction network (x_0, referred to as a "target cell"). Note that this objective functions allows back-propagation of the gradients through both the denoising decoder p_θ and the sample encoder f_ϕ.

The training procedure without mini-batching is described in detail in Algorithm 1. For training, we use the AdamW optimizer with learning rate 1×10^{-3}. Gradients are clipped at a threshold of 0.1.

Algorithm 1 Conditional Diffusion Autoencoding Training

Input: Encoder f_ϕ, noise predictor ϵ_θ, time steps T, number of anchor cells N_{anchor}, number of target cells N_{target}, number of time steps per sample N_{time}, inverse variance schedule $(\bar{\alpha}_t)_{t=1}^T$, patient samples $(X^j)_{j=1}^M$.
repeat
  Sample patient $j \sim \mathcal{U}([M])$
  Subsample N_{anchor} anchor cells $A \subset X^j$ and N_{target} target cells $Y \subset X^j$
  Sample N_{time} time steps $t_\ell \sim \mathcal{U}([T])$
  Compute patient representation $z = f_\phi(A)$
  Sample $N_{\text{target}}N_{\text{time}}$ noise vectors $\epsilon_{k\ell} \sim \mathcal{N}(0, I)$
  Compute noisy cells $\tilde{y}_{k\ell} = \sqrt{\bar{\alpha}_{t_\ell}}y_k + \sqrt{1 - \bar{\alpha}_{t_\ell}}\epsilon_{k\ell}$
  Compute loss $L(\phi, \theta) = \sum_{k=1}^{N_{\text{target}}} \sum_{\ell=1}^{N_{\text{time}}} \|\epsilon_{k\ell} - \epsilon_\theta(\tilde{y}_{k\ell}, t_\ell, z)\|^2$
  Update θ and ϕ using $\nabla_{\theta,\phi} L$
until loss has converged

5.3.4 Interpretation of scSet as an autoregressive model

Our work shares a conceptual connection with autoregressive modeling. Viewing each cell in a sample as a token similar to those used in natural language processing, scSet aims to learn the joint distribution over tokens, for which autoregressive approaches have recently shown great capabilities [157]. However, autoregressive models typically process sequences, where there is a canonical ordering that determines the next token to predict. Further, autoregressive approaches generally learn a probability distribution over discrete tokens, not continuous

samples such as single-cell profiles. Inspired by the framework introduced in Li et al. (2024) [158], scSet can be seen as an autoregressive model which selects an arbitrary ordering of cells for each sample, and then uses the first $N_{\text{anchor}} < N$ cells to predict a latent prototype of the next cell state, namely the patient representation, which is subsequently transformed into an actual cell state by the diffusion model.

5.4 Experiments

We evaluate scSet's learned patient embeddings through (i) qualitative and quantitative evaluations of the unsupervised trained embeddings and (ii) using the patient embeddings as input to downstream clinical prediction tasks. We describe the datasets, metrics, baselines and results for each of these approaches in Sections 5.4.1 and 5.4.2, respectively.

5.4.1 Training Patient Representations via Conditional Diffusion

In this section, we validate that the patient representations learned by scSet capture known variations between patients. First, we show that the diffusion model decodes the expected distribution of cell types for each patient, and then we use real and semi-synthetic data to validate that patients with known differences are separated in the latent space.

Data

The scSet autoencoder was trained on data from the CZ CELLxGENE Discover Census [159], containing 7,342 samples (after filtering for samples with at least 128 cells) from diverse tissue and disease contexts. scVI embeddings provided in the Census were used to represent input cells; we retained 14 scVI latents by filtering to latents with standard deviation $> .4$ across the pretraining corpus. We chose to represent cells using scVI embeddings rather than raw gene expression as it already partially corrects for batch effects and reduces the dimensionality of inputs. We used 90% of patients for training and held out 10% for evaluation.

For our semi-synthetic data, we created patient samples by resampling cells from a pool of 8064 immune cells (natural killer cells, helper T cells, CD8+ cytotoxic T cells, and monocytes) from 32 patients from a multiple myeloma study [60] that was not part of the pretraining corpus, in order to create synthetic patients belonging to different synthetic "patient subtypes." For each subtyping experiment, we simulated 12 patients, each with 200 cells. For our cell type composition experiment, we created samples that were enriched for a given cell type: we randomly sampled cells of the dominant cell type to account for 55% of the sample composition, and the remaining cell types to each account for 15%. For

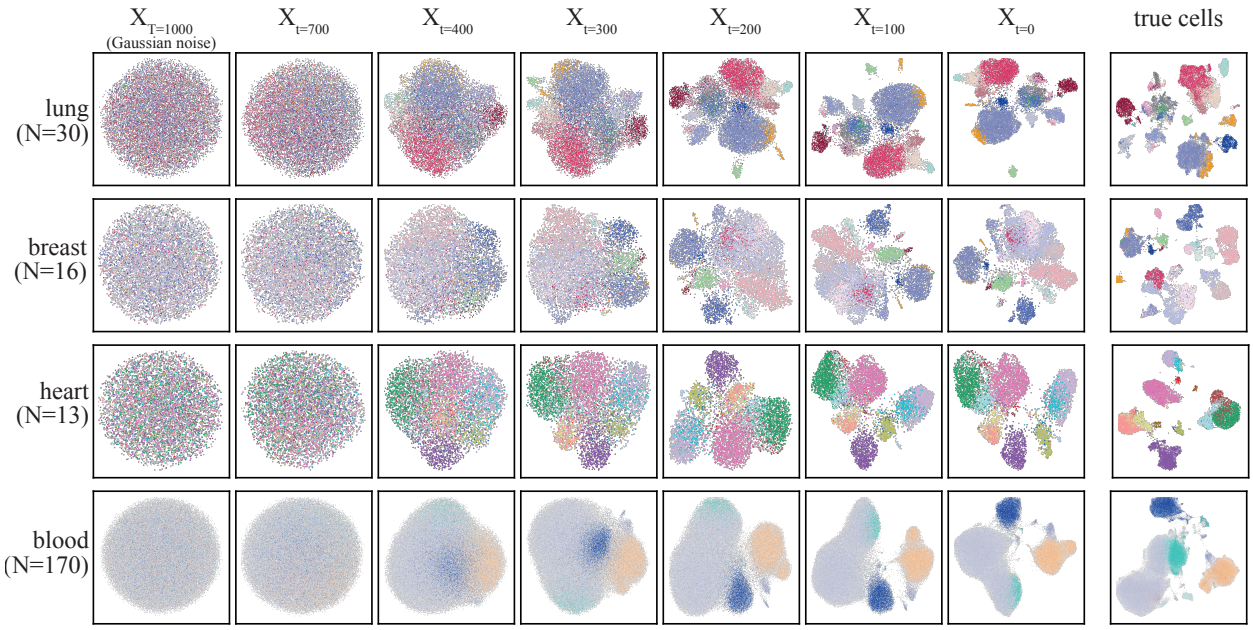


Figure 5.2: UMAP visualizations of cells over timesteps of the denoising process, starting from random noise ($X_{T=1000}$). Test set patient samples were embedded using f_ϕ and used to condition the denoising network; the union of these cells is shown in the right column, labeled "true cells." 500 cells per patient were generated. For visualization purposes, each row contains the union of cells from a given tissue (N = number of patients per tissue). True cells are colored by their ground-truth cell type and simulated cells are colored by pseudo-cell type labels, obtained by predicting cell types using a k-Nearest Neighbors classifier trained on all cells in the test set.

our perturbation subtyping experiment, we created a "perturbed" subtype in which cell types were present in equal proportions to their unperturbed counterparts, but with a slight phenotypic shift in helper T cells, and an equal and opposite shift in CD8+ cytotoxic T cells. Specifically, we added a constant to five of the latent dimensions for all helper T cells, and subtracted this same constant from the same latents for CD8+ cytotoxic T cells in those same samples. We chose this perturbation structure because it represents a case where averaging, or pseudobulking, the sample would not be able to pick up on this shift due to the equal and opposite effect of the two perturbations.

Results

If the model has learned meaningful patient representations, we expect the diffusion decoder—which is conditioned on those representations—to generate sets of cells that closely resemble a patient's true cells. Thus, we ran inference on the evaluation set (10% of patients that were not used to train scSet), decoding 500 cells per patient (the number of decoded cells is arbitrarily set by the user). Starting from Gaussian noise at time step $T = 1000$, we

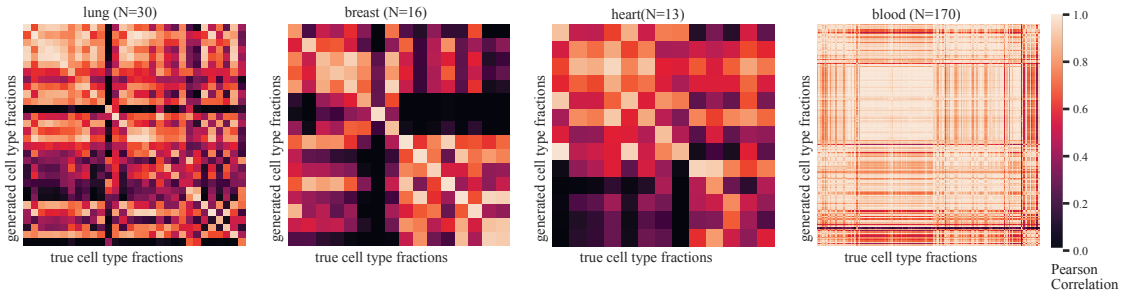


Figure 5.3: Pearson correlations between true and reconstructed cell type proportions for evaluation set patients from each tissue (the same patients shown in Figure 5.2). scSet reconstructs a set of cells per patient which matches the relative proportions of cell types in the true patient sample.

visualize via UMAP [160] the reconstructed cell profiles of patients grouped by tissue as they are denoised over time steps in the diffusion decoder (Figure 5.2). Color-coding cells by their cell types shows that for each tissue, the model generates the same cell types which were present in the ground truth single-cell data, and in relatively similar proportions (the Pearson correlation coefficient between the true and reconstructed cell type proportions for each tissue was consistently high: 0.95 for lung, 0.97 for breast, 0.89 for heart, and 0.91 for blood). We include tables showing the true and reconstructed cell type proportions for each tissue in Table 5.1, Table 5.2, Table 5.3, and Table 5.4. Note that since generated cells have no ground truth cell type labels, we predicted instead pseudo-cell type labels from their simulated profiles at $t = 0$ using a k-Nearest Neighbors classifier trained on the true cells from these patients.

While Figure 5.2 and its associated tables show that the landscape of true cells for each tissue matches the landscape of generated cells, we wanted to confirm that the patient representations condition the model to generate *patient-specific* profiles, rather than simply tissue-specific profiles. We calculated the Pearson correlations between the cell type proportions vectors of the true and generated cells for each sample in a tissue, and observed that the generated cell type distribution for a given patient is usually most strongly correlated with the ground-truth distribution of cell types in the patient used for conditioning. We visualize these results in Figure 5.3.

Finally, we qualitatively validate that the patient representations learned are reasonable. First, we inspect the embeddings for 1,500 patient samples from the CELLxGENE Discover Census. Coloring each patient sample by its tissue type, Figure 5.4 reveals that scSet representations separate patient samples by their tissue origin, as expected. We next ran hierarchical clustering on patient embeddings from our semi-synthetic data, and observed that scSet can separate patients based on differences in cell type proportions (Figure 5.5a) as

Table 5.1: Cell type proportions for true and scSet reconstructed **lung** samples. Limited to 30 most common cell types among true cells.

cell type	synthetic proportions	true proportions
macrophage	0.201715	0.201773
T cell	0.261124	0.187556
monocyte	0.119152	0.088269
type II pneumocyte	0.033254	0.071061
endothelial cell	0.043735	0.068700
fibroblast	0.042020	0.045328
natural killer cell	0.019104	0.044489
ciliated columnar cell of tracheobronchial tree	0.014721	0.040659
B cell	0.020867	0.038324
epithelial cell	0.064459	0.033209
dendritic cell	0.029967	0.028383
malignant cell	0.013530	0.026704
type I pneumocyte	0.010624	0.023661
neutrophil	0.005860	0.022848
plasma cell	0.004288	0.021982
mast cell	0.014960	0.017470
smooth muscle cell	0.003954	0.005168
pericyte	0.002620	0.004827
secretory cell	0.003859	0.004695
club cell	0.001810	0.004197
myeloid cell	0.002573	0.004013
nasal mucosa goblet cell	0.001906	0.003830
respiratory basal cell	0.000858	0.003095
lung ciliated cell	0.000048	0.002885
lung pericyte	0.000238	0.002308
lung goblet cell	0.000048	0.002020
neuron	0.000286	0.000944
ciliated cell	0.016246	0.000577
stromal cell	0.000667	0.000367
mesothelial cell	0.001048	0.000341

Table 5.2: Cell type proportions for true and scSet reconstructed **blood** samples. Limited to 30 most common cell types among true cells.

cell type	synthetic proportions	true proportions
T cell	0.671453	0.407475
naive T cell	0.017003	0.168529
monocyte	0.161800	0.156635
natural killer cell	0.030631	0.122473
B cell	0.101011	0.064810
naive B cell	0.003179	0.040356
dendritic cell	0.011306	0.011751
platelet	0.000768	0.006774
plasmablast	0.000804	0.003751
T-helper	0.000009	0.003279
blood cell	0.000357	0.003261
T follicular helper cell	0.000071	0.002978
progenitor cell	0.000911	0.001455
erythrocyte	0.000170	0.001065
lymphocyte	0.000143	0.001038
thymocyte	0.000054	0.000916
plasma cell	0.000036	0.000687
IgG plasma cell	0.000009	0.000445
IgA plasma cell	0.000009	0.000427
double negative T regulatory cell	0.000009	0.000337
innate lymphoid cell	0.000036	0.000296
macrophage	0.000054	0.000225
IgA plasmablast	0.000161	0.000180
common lymphoid progenitor	0.000009	0.000180
megakaryocyte	0.000000	0.000166
ILC1, human	0.000000	0.000126
myeloid cell	0.000000	0.000117
IgM plasma cell	0.000000	0.000108
IgG plasmablast	0.000009	0.000085
megakaryocyte-erythroid progenitor cell	0.000000	0.000076

Table 5.3: Cell type proportions for true and scSet reconstructed **breast** samples. Limited to 30 most common cell types among true cells.

cell type	synthetic proportions	true proportions
epithelial cell	0.270300	0.204534
fibroblast	0.139667	0.146364
T cell	0.158140	0.115783
endothelial cell	0.073285	0.092961
progenitor cell	0.079273	0.089614
basal cell	0.070646	0.079420
endothelial tip cell	0.070443	0.060402
luminal hormone-sensing cell of mammary gland	0.024259	0.046506
perivascular cell	0.017052	0.036515
pericyte	0.009541	0.022112
macrophage	0.046488	0.021452
smooth muscle cell	0.002436	0.018359
subcutaneous adipocyte	0.007917	0.014657
B cell	0.006293	0.008317
plasmablast	0.001827	0.007151
naive B cell	0.000812	0.005376
natural killer cell	0.001827	0.004970
monocyte	0.007714	0.004564
IgA plasma cell	0.000812	0.004311
dendritic cell	0.006192	0.003398
naive T cell	0.000000	0.002891
lymphocyte	0.000000	0.002434
myeloid cell	0.001320	0.002384
Tc1 cell	0.000000	0.001471
leukocyte	0.000000	0.001065
mast cell	0.002944	0.001014
contractile cell	0.000406	0.000710
IgG plasma cell	0.000203	0.000710
neutrophil	0.000203	0.000558

Table 5.4: Cell type proportions for true and scSet reconstructed **heart** samples. Limited to 30 most common cell types among true cells.

cell type	synthetic proportions	true proportions
fibroblast	0.232488	0.193614
endothelial cell	0.117873	0.151561
regular ventricular cardiac myocyte	0.144636	0.133050
pericyte	0.066325	0.113281
cardiac muscle cell	0.086572	0.110705
mural cell	0.033279	0.098604
regular atrial cardiac myocyte	0.082965	0.066016
myeloid cell	0.085408	0.046247
macrophage	0.072609	0.020128
smooth muscle cell	0.001280	0.018211
lymphocyte	0.041541	0.015755
T cell	0.023272	0.006110
cardiac neuron	0.001629	0.005931
epicardial adipocyte	0.002444	0.005871
monocyte	0.001513	0.005152
neural cell	0.001164	0.005092
mast cell	0.003258	0.001677
natural killer cell	0.000000	0.000839
dendritic cell	0.001745	0.000779
adipocyte	0.000000	0.000719
mesothelial cell	0.000000	0.000659

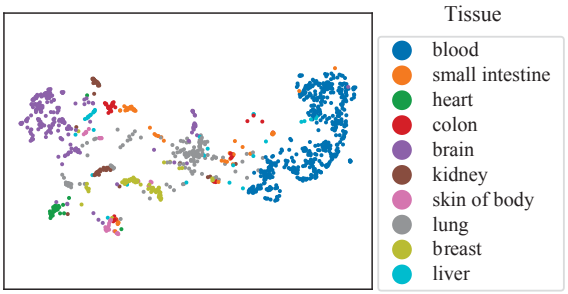


Figure 5.4: UMAP visualization of the patient embeddings encoded via scSet, colored by tissue type.

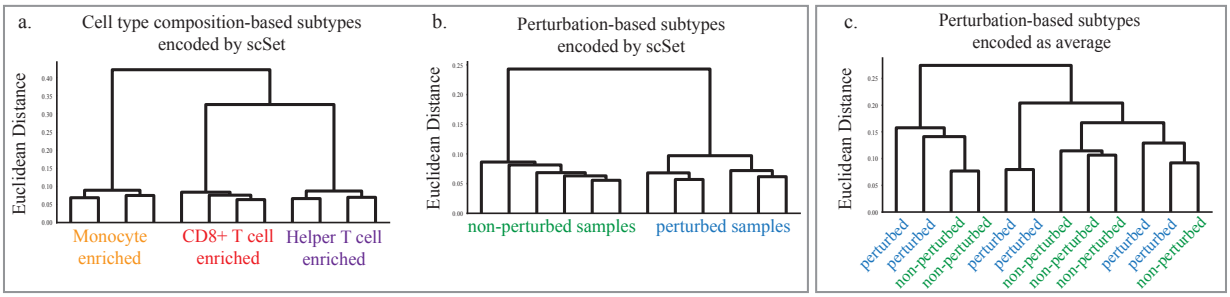


Figure 5.5: Hierarchical clustering of semi-synthetic samples that were generated as part of the (a) cell type composition subtype experiment or (b) cell type perturbation subtype experiment. The left box shows that scSet embeddings for these semi-synthetic patients cluster by subtype. The right box (c) shows that a simple AVERAGE embedding of the perturbed patients do not cluster by subtype.

well as shifts in cell states, or phenotypes (Figure 5.5b). For the perturbation experiment, we intentionally induced a perturbation that could not be detected between samples whose cells had simply been averaged (since equal and opposite perturbations were imposed on different cell types), highlighting that scSet can capture signal in its patient embeddings that pseudobulk would not be able to (Figure 5.5c).

5.4.2 Clinical Prediction from Patient Representations

We evaluated our learned embeddings by using them as input to supervised models for predicting patient-level phenotypes for the datasets described below in Section 5.4.2. Our clinical prediction models are comprised of scSet's transformer encoder, which aggregates single-cell data into a patient-level representation via the `[CLS]` token, and an appended prediction head, as described in Section 5.4.2.

Data & Tasks

HLCA. The human lung cell atlas (HLCA) [161] combines 49 datasets related to the human respiratory system, integrating over 2.4 million cells from 486 individuals. In our "HLCA triple" task, we train models to discriminate between the three most prevalent disease states for lung tissue samples in this dataset: normal (n=216 samples), COVID-19 (n=82), and pulmonary fibrosis (PF) (n=71). We use a 10-fold cross validation scheme and assign all the patients from a given dataset to the same fold to avoid confounding by dataset-specific batch effects. This setup requires the model to generalize across batches, and is significantly more challenging than an ungrouped K-fold scheme, but better reflects a potential real-world deployment setting for our model. We include results from a binary version of this task, discriminating between normal and PF patients, in Appendix B.1.

SLE. The systemic lupus erythematosus (SLE) dataset [162] contains 1.2 million peripheral blood mononuclear cells (PBMCs) from 162 patients with SLE and 99 healthy controls. We train models to discriminate between SLE and healthy samples, and use a standard 10-fold cross-validation scheme for evaluation.

COVID-19. The COVID-19 dataset [163] profiles transcriptomes of 624,325 peripheral blood mononuclear cells from 24 healthy donors and 102 patients with varying severities of COVID-19, ranging from asymptomatic to critical disease. We train models to discriminate between COVID-19 and healthy samples, and use a standard 10-fold cross-validation scheme for evaluation.

Each dataset was pre-processed to embed cells using the trained scVI model available on CELLxGENE Census. The same 14 latent dimensions as used for pretraining scSet were retained.

Baseline patient encoders

We compared our transformer-based encoder to multiple baseline encoders with varying degrees of complexity: (i) Average takes a simple average of features across cells, as is currently common practice when summarizing scRNA-seq to the patient level. (ii) Cell type fractions and (iii) Cell type means summarize cell type level information, either the fractions of cells in the sample assigned to each cell type, or the average of features for cells of a given specific cell type, concatenated together for all cell types in the dataset. (iv) Cell type fracs+means is the concatenation of the two. These cell type level summary vectors have been shown to correlate with clinical features [132, 133], but require expert

labeling of cell types in order to construct, thus representing an expert-engineered baseline. (v) KMEANS cluster data into K clusters and concatenate the mean embeddings from each cluster. The K-means is trained on the training set, with results from K=30 and K=60 shown here. (v) Inspired by recent work from [164], SCSET W/ FLOW DECODER uses a flow-based decoder during pretraining instead of a diffusion model. (vii) Finally, we compare to another attention-based encoder, ABMIL [139], as recent work for supervised clinical prediction from scRNA-seq employed this architecture [134, 139]. Of note, while ABMIL uses attention to compute a parameterized weighted average of cells, it does not compute *self-attention* between input cells as our transformer encoder does.

To tease apart the benefit afforded by the architecture of the encoder vs. our diffusion pretraining, we included ablation baselines for each of the parametric encoders, evaluating the performance of scSet and ABMIL encoders both with and without diffusion pretraining. This ablation is conceptually similar to that which we ran for scBERT and scGPT in Chapter 4 (Section 4.2.3), and we believe this is an important ablation to run for any pre-trained foundation model.

Prediction models.

We input the patient embeddings to 3 different prediction models: (i) LINEAR PROBE, an L2-regularized logistic regression model, (ii) MLP, a simple multilayer perceptron with 2 hidden layers and GELU activations [165], and (iii) FINETUNE END-TO-END (FT-E2E), which uses the MLP from (ii) but jointly finetunes the encoder and MLP end-to-end, allowing gradient updates to propagate through both components to adapt the encoder's representations for the downstream task.

For the MLP-based prediction heads, we use a weighted cross-entropy loss to compensate for class imbalance. Hyperparameter tuning was performed using nested K-fold validation, with inner K=5 and outer K=10, as described in Section 5.4.2. Hyperparameter details are provided in Section 5.7.1.

Metrics

We report the F1 score (↑) across folds, which balances precision and recall, making it suitable for class-imbalanced settings. For multiclass tasks, we use the weighted F1 score, averaging per-class F1 values weighted by the number of positives. Accuracy and AUC scores are reported in Appendix B.1.

Table 5.5: Performance of scSet and baselines on clinical prediction tasks, as described in Section 5.4.2. Average weighted F1-Scores across folds ± SEM are shown. Best performers (by mean) are bolded.

Task	COVID binary			HLCA triple			SLE		
Supervised model	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E
scSet	**.95±.02**	**.93±.02**	**.93±.02**	.78±.06	.68±.07	**.66±.08**	.93±.02	**.94±.01**	**0.95±0.01**
scSet w/o diffusion	.9±.03	.86±.02	.88±.03	.61±.06	.47±.08	.53±.09	.92±.02	.83±.04	.92±.02
scSet w/ flow decoder	.9±.03	.71±.04	.87±.03	.57±.07	.43±.09	.57±.07	.87±.02	.76±.04	.92±.01
ABMIL w/ diffusion	.87±.03	.83±.03	.85±.03	.62±.08	.57±.06	.52±.05	.87±.02	.9±.01	.92±.01
ABMIL w/o diffusion	.88±.03	.86±.03	.84±.03	.57±.07	.38±.07	.52±.06	.87±.02	.82±.01	.9±.02
average	0.88±.02	.8±.04	.81±.04	.58±.07	.49±.06	.5±.05	.88±.02	.75±.03	.77±.03
cell type fractions	.84±.02	.68±.04	N/A	.6±.07	.51±.09	N/A	.83±.02	.72±.03	N/A
cell type means	.87±.03	.92±.03	N/A	.73±.04	.68±.04	N/A	.94±.01	.86±.03	N/A
cell type fracs+means	.87±.03	.92±.03	N/A	.72±.04	**.7±.05**	N/A	**.95±.01**	.9±.02	N/A
kmeans30	.92±.03	.86±.04	N/A	.77±.05	.63±.06	N/A	.94±.02	.92±.02	N/A
kmeans60	.94±.02	.9±.02	N/A	**.79±.04**	.66±.06	N/A	.89±.02	.92±.01	N/A

Results

Across most tasks and prediction models, scSet outperforms all other encoders and ablation models (Table 5.5). Our ablation baselines (scSet w/o diffusion and ABMIL w/o diffusion) suggest that pretraining the encoder via our conditional-diffusion autoencoder improves downstream supervised performance. Some of the strongest baselines were the cell type-level summary vectors (Cell type means or Cell type fracs+means), which require expert annotation of cell type labels to construct, making scSet a competitive alternative that does not require cell type labels. The Kmeans baseline, which does not require expert labels, was also quite strong, and as such, deep foundation models will need to show utility over this baseline in order to warrant widespread adoption.

In real-world settings, clinically-labeled scRNA-seq cohorts are often small [159], and thus a model that can improve predictive performance on small amounts of labeled data is valuable. With this in mind, we evaluated each model's performance when trained on just 25, 50, or 100 training samples per-fold. We repeated this experiment five times, each time using a different random subset of data, and we report the mean and standard error of the mean (SEM) across all random subsets and test folds. Even with limited training data, scSet consistently outperforms the baseline models (Figure 5.6).

5.5 Discussion

Our results demonstrate that scSet effectively learns meaningful patient-level representations from single-cell RNA sequencing data through a diffusion-based autoencoding framework. By leveraging a transformer-based encoder to aggregate unordered single-cells, and employing a conditional diffusion decoder to generate realistic cellular compositions, scSet provides a pow-

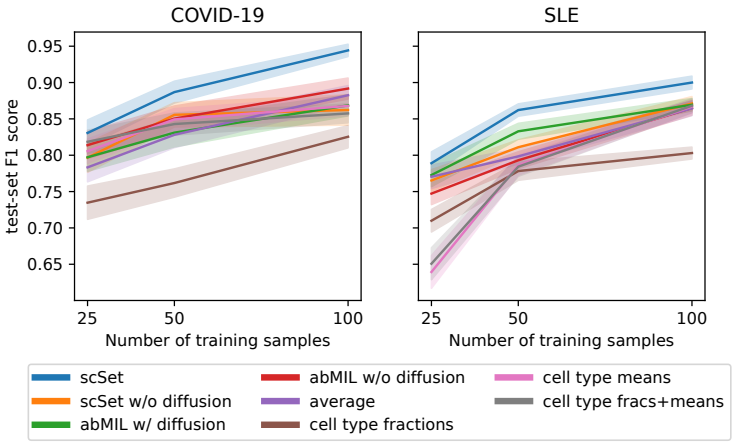


Figure 5.6: With limited numbers of training samples, scSet still outperforms baseline encoders on the COVID-19 and SLE prediction tasks. Error bars represent the standard error, calculated over 5 random subsamplings of the training data for each of 10 folds.

erful and flexible method for patient-level modeling that elegantly circumvents the challenge of autoencoding set-structured data. scSet embeddings prove useful for downstream clinical prediction tasks, suggesting that scSet captures clinically relevant signals that generalize across datasets. While we explored non-deep baselines which were also strong, such as the KMEANS baseline, foundation models like scSet may provide improved patient embeddings in terms of learning meaningful distances in the latent space and offering greater opportunities for biological discovery through model introspection. Future work should more deeply probe the relative benefits and costs of scSet compared to other strong baselines.

The introduction of self-supervised learning for patient-level representations from scRNA-seq data mitigates the common issue of limited labeled datasets in biomedical applications. By learning from large-scale unlabeled data, scSet can create pretrained representations that transfer effectively to new clinical prediction tasks with minimal labeled data. This approach is particularly advantageous for studying rare diseases or heterogeneous conditions where labeled single-cell samples are scarce. Additionally, our framework is modular and can incorporate different cell embeddings, making it adaptable to future advances in single-cell representation learning.

5.6 Limitations and Future Work

While scSet presents a promising framework for patient-level representation learning, several limitations remain. First, our current model is trained on cells represented by precomputed scVI embeddings. While this allows for standardized inputs across datasets and mitigates

batch effects, future work could explore end-to-end training of cell and patient embeddings, potentially improving interpretability and performance.

Further analysis is needed to assess the contribution of the scVI cell embeddings to the expressivity of the learned patient representations. We observed that conditioning the denoising network on the *average* scVI embedding in a sample also enables realistic cell reconstruction. While we confirmed that a transformer encoder provides benefits for downstream clinical prediction tasks, future work could investigate whether a transformer encoder provides significant advantages over simpler encoding schemes for the purposes of single-cell simulation.

Our current approach conditions the generative diffusion model on patient representations derived from scRNA-seq data, but this framework could be extended to generate single-cell profiles conditioned on bulk RNA-seq profiles, patient characteristics, or single-cell data from other modalities.

Additionally, as spatial scRNA-seq data becomes more widely available, future extensions could integrate spatial information, which is often predictive of clinical outcomes [128, 166].

Finally, scSet and similar methods will be most valuable if they can offer biological insight, rather than just black-box clinical prediction. To this end, future work shoud focus on model interpretability, which has the potential to illuminate novel cell types and cell states that are correlated with clinical outcomes.

5.7 Supplementary Methods

5.7.1 Hyperparameter tuning

For our scset model, which is made up of a transformer and a denoising diffusion network, we used the following hyperparameters for the transformer: 4 transformer heads, 2 blocks (layers) of transformers, batch size 32, and learning rate 10^{-3}. We searched over the following hyperparameters: number transformer heads $\{2, 4\}$; number transformer blocks $\{2, 3, 4\}$; batch size $\{16, 32, 64, 128\}$; learning rate $\{10^{-2}, 10^{-3}, 10^{-4}\}$. We found that our choice of batch size and learning rate significantly affected validation set denoising loss. While models trained with different batch sizes converged to a similar loss by 200 epochs, we found that the larger the batch size, the longer the model took to converge to this loss. We chose a batch size of 32, to balance this behavior (which would suggest choosing the lowest batch size) with efficient use of our GPUs (which are only partially utilized at lower batch sizes). Once the batch size was fixed, a learning rate of 10^{-3} performed best. The number of transformer heads and blocks did not meaningfully alter performance, so we settled on 4 heads and 2

blocks to balance expressivity with avoiding overfitting and unnecessary complexity.

For the logistic regression in sklearn, we tuned the hyperparameter C over the following values using nested cross-validation: [0.01, 0.1, 1, 10, 100, 1000, 10000, 100000, 1000000].

5.7.2 Compute environment

Models were trained on a single NVIDIA A100 80GB GPU, on a cluster with 504GB RAM.

5.7.3 Statistics

We ran each task on K held out test folds. We report the 95% confidence intervals for the mean performance across these folds. To calculate these intervals, we determined the sample mean (x) and sample standard deviation (s) for the performance metrics, then computed the standard error of the mean (SEM) as $s/\sqrt(n)$, where n is the number of runs (n=10). For n=10, the t-value for a 95% confidence level is 2.262. The margin of error (ME) was obtained by multiplying the t-value with the SEM. We reported 95% confidence intervals as x±ME.

5.8 Code Availability

Software and analysis code are publicly available at https://github.com/clinicalml/scset-2025.

Chapter 6

Discussion

In this thesis, we explored and developed machine learning methods that enable biological discovery and clinical predictions from scRNA-seq data. We first opened in Chapter 2 by analyzing scRNA-seq data from a cohort of patients with multiple myeloma and its precursor conditions. In addition to uncovering biomarkers at very early stages of disease, we gained insight into the signal contained in scRNA-seq data as well as the challenges that hinder deeper analysis, which we subsequently addressed through machine learning approaches. A significant challenge identified in this initial analysis was that cell representations were dominated by patient-specific signal, such that even cells from patients with similar clinical disease subtypes were transcriptionally distinct, making it challenging to discover shared cellular phenotypes across patients. We hypothesized that copy number variations may be the source of these wide-scale transcriptional differences between cells from different patients, and in Chapter 3 we introduce a generative model that disentangles CNV-driven expression from other biological variation, enabling more interpretable, robust downstream analyses and facilitating cross-patient comparisons. Subsequently, in Chapter 4, we investigated the emerging class of single-cell foundation models that purport to learn richer cell representations than traditional methods. Our findings revealed that current benchmarking tasks for evaluating these cell representations lack sufficient complexity, as linear methods performed comparably to deep foundation models. These results caution against premature adoption of complex models and emphasize the necessity for rigorous benchmarking protocols. Finally, Chapter 5 presents scSet, a transformer-based framework for patient-level prediction from scRNA-seq. scSet introduces a self-supervised training approach that learns representations of patients based on their constituent cells, thereby enabling outcome prediction even in settings with limited labeled data. Collectively, these contributions are part of a broader vision: to design ML models that are biologically grounded and ultimately capable of translating single-cell insights into clinical action.

Looking to the future, thoughtfully designed foundation models that learn cell and patient representations have potential to significantly accelerate biological discovery, disease diagnosis and prognosis, personalized medicine, drug discovery, and drug design. However, a substantial gap persists between the current capabilities of these models and their potential real-world impact.

Current architectures and training objectives for foundation models are predominantly adapted from other domains such as natural language processing. The development of models specifically tailored to biological data may improve performance and interpretability. For example, the noise structure and technical artifacts present in scRNA-seq data differ fundamentally from those in text data, necessitating specialized methods for noise reduction. Furthermore, whereas words represent categorical features, gene expression manifests as count values or, after normalization, continuous values. The tokenization strategy for gene expression prior to modeling can significantly impact model efficacy and warrants comprehensive exploration. Additionally, a crucial distinction between scRNA-seq data and text lies in the absence of inherent sequential structure in the former, both between genes in a cell (whose relationships are modeled when learning cell representations) and between cells in a dissociated patient biopsy (whose relationships are modeled when learning patient representations). Nevertheless, certain models such as scGPT [8] employ autoregressive causal masking for training attention layers, a technique developed for sequential data. The efficacy of autoregressive modeling for scRNA-seq merits deeper investigation. As the field matures, the development of models which have been crafted specifically for biological data may offer enhanced biological insight. As one example, graph based models [167] may be well suited to capture the underlying relationships between cells and genes, and should be explored in the future.

The quality of artificial intelligence systems fundamentally depends on their training data. The challenge of limited labeled data is particularly acute in patient-level representation learning, as much of the available transcriptional data lacks consistent clinical outcome annotations. Single-cell datasets with clinical outcome annotations remain scarce and are often siloed in independent clinical trials, many of which are not publicly accessible. In Chapter 4, we addressed this challenge through a self-supervised approach to patient representation learning. However, the development and release of large-scale, consistently annotated clinical single-cell datasets would substantially enhance our capacity to extract patient-level insights. Additionally, the integration of multimodal data—encompassing genomics, transcriptomics, epigenomics, imaging, and clinical metadata—represents an exciting frontier for future research. Furthermore, scaling laws for single-cell foundation models remain undetermined, and the relationship between model expressivity, performance, and training dataset size

constitutes an important area for investigation.

Robust benchmark tasks and datasets are essential for advancing single-cell foundation models. As discussed in Chapter 4, the specific tasks that will benefit most—if at all—from single-cell foundation models remain unclear. Our results demonstrated that linear models are competitive with current foundation models for cell type annotation, and similar findings have been reported for predicting cellular responses to perturbation [168, 169]. The application of single-cell foundation models to decode gene regulatory networks represents a promising direction for future research. However, since the ground truth gene regulatory network remains unknown, the field must develop compelling benchmarks to measure progress effectively. For both cell- and patient-level scRNA-seq foundation models, the field requires standardized benchmark datasets with challenging tasks where simple models underperform, and with pre-defined test sets comprising patients and studies not encountered during training. Such benchmarks will facilitate clear and robust evaluation of model performance, generalization capabilities, and clinical relevance.

The research presented in this thesis aims to develop methods that harness the rich signal contained in patient data at single-cell resolution to yield biological discovery and clinical insights. By developing methods that span from patient-specific biomarker discovery to self-supervised patient representation learning, we have helped shape a growing discipline at the intersection of single-cell genomics and clinical machine learning. These advances, while significant, represent initial steps toward the ultimate goal of translating high-dimensional molecular data into actionable clinical insights that improve patient outcomes and transform precision medicine.

Appendix A

Supplementary Data for Chapter 2

Table A.1: A mapping between the sample IDs for the CD138+ cells analyzed in this study and the corresponding sample IDs for the CD138- cells analyzed in Zavidij et al.

Zavidij2020_IDs	**Boiarsky2022_IDs**
NBM-1	NBM-1
NBM-2	NBM-2
NBM-3	NBM-3
NBM-4	NBM-4
NBM-5	N/A
N/A	NBM-6
N/A	NBM-7
NBM-8	NBM-8
NBM-9	N/A
NBM-10	NBM-10
NBM-11	NBM-11
MGUS-1	MGUS-1
MGUS-2	MGUS-2
MGUS-3	MGUS-3
MGUS-4	MGUS-4
N/A	MGUS-5
MGUS-6	MGUS-6
SMMl-1	SMM-1
SMMl-2	SMM-2
SMMl-3	SMM-3
N/A	SMM-4
SMMh-2	SMM-5
SMMh-3	SMM-6
SMMh-4	SMM-7
N/A	SMM-8
SMMh-6	SMM-9
SMMh-7	N/A
SMMh-8	SMM-10
SMMh-9	SMM-11
SMMh-10	SMM-12
MM-1	MM-1
N/A	MM-2
MM-3	MM-3
MM-4	MM-4
MM-5	MM-5
MM-6	MM-6
MM-7	MM-7
MM-8	MM-8

Table A.2: List of clinical measurements for samples used for single-cell RNA sequencing, including age, disease stage, sex, race, Type (type of the immunoglobulin involved in myeloma), M Protein (g/dL), BMPC % (% plasma cells in bone marrow biopsy), serum free light chain ratio (involved/uninvolved), progression to MM (1=Has progressed to MM, 2=Has not progressed to MM, 3=MM was the original diagnosis), days to MM diagnosis (from the time of initial diagnosis), treated during MGUS/SMM (0=no, 1=yes), and follow up time per patient (days).

sample_ID	age (range)	disease_stage	sex	race	Type	M Protein (g/dL)	BMPC %	serum free light chain ratio (involved/uninvolved)	progression to MM	days to MM diagnosis	treated during MGUS/SMM	follow up time (days)
NBM-1	30-39	NBM	female	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-2	30-39	NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-3	50-59	NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-4	40-49	NBM	female	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-6	60-69	NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-7	30-39	NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-8	60-69	NBM	female	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-10	40-49	NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
NBM-11		NBM	male	nan	nan	nan	nan	nan	nan	nan	nan	nan
MGUS-1	80-89	MGUS	male	White	IgG Lambda	0.44	<5%	0.54	2		0	1534
MGUS-2	40-49	MGUS	male	White	IgG Kappa	0.54	Around 10%	1.84	2		0	5314
MGUS-3	60-69	MGUS	female	White	IgG Kappa	0.85	7.50%	1.53	2		0	1425
MGUS-4	60-69	MGUS	female	White	IgG Kappa	Faint	<5%	1.43	2		0	1491
MGUS-5	60-69	MGUS	female	White	IgG Kappa	0.7	7.50%	1.37	2		0	1838
MGUS-6	60-69	MGUS	female	White	IgG Lambda	0.67	7.50%	0.68	2		1	3025
SMM-1	60-69	SMM	male	White	IgG Kappa	2.35	20%	61.79	1	1476	0	2796
SMM-2	60-69	SMM	female	White	IgG Kappa	1.52	12.50%	3.29	2		0	4006
SMM-3	50-59	SMM	male	White	IgG Kappa	0.93	20%	3.69	2		0	3062
SMM-4	50-59	SMM	male	White	IgA Kappa	0.43	11%	1.52	2		1	3125
SMM-5	50-59	SMM	female	White	IgG Kappa	2.74	30%	4.68	2		1	2188
SMM-6	60-69	SMM	male	White	IgG Kappa	3.51	50%	16.49	1	321	1	1895
SMM-7	50-59	SMM	female	White	IgA Kappa	2.94	35%	11.94	1	2859	1	3868
SMM-8	50-59	SMM	male	White	IgA Kappa	0.31	10%	11.89	1	1393	1	1659
SMM-9	40-49	SMM	male	White	IgA Kappa	0.36	20%	3.03	2		1	1738
SMM-10	80-89	SMM	male	White	IgG Lambda	0.91	12.50%	52.66	1	2186	1	2446
SMM-11	60-69	SMM	male	White	IgG Kappa	1.69	35%	66.79	1	120	0	1565
SMM-12	70-79	SMM	female	White	IgG Kappa	0.69	75%	219.29	1	423	1	1921
MM-1	60-69	MM	male	White	IgA Lambda	0.45	100%	1,406.13	3	0	nan	2246
MM-2	50-59	MM	male	White	IgG Kappa	4.71	28%	80.64	3		nan	1783
MM-3	70-79	MM	male	White	IgG Kappa	2.64	25%	24.97	1	2591	nan	4912
MM-4	70-79	MM	male	White	Lambda light chain	0.23	40%	165.18	3	0	nan	1400
MM-5	60-69	MM	male	Hispanic	IgG Kappa	4.15	65%	2.44	3	0	nan	1748
MM-6	60-69	MM	female	White	IgG Kappa	4.35	40%	4.00	3	2	nan	1765
MM-7	50-59	MM	female	White	IgG Kappa	2.99	50%	1.62	3	426	nan	2130
MM-8	60-69	MM	male	White	IgG Kappa	3.06	25%	87.47	1	1476	nan	2796
MGUS-2	40-49	MGUS	male	White	IgG Kappa	0.54	Around 10%	1.84	2		0	5314
MGUS-3	60-69	MGUS	female	White	IgG Kappa	0.85	7.50%	1.53	2		0	1425
MGUS-4	60-69	MGUS	female	White	IgG Kappa	Faint	<5%	1.43	2		0	1491
MGUS-5	60-69	MGUS	female	White	IgG Kappa	0.7	7.50%	1.37	2		0	1838
MGUS-6	60-69	MGUS	female	White	IgG Lambda	0.67	7.50%	0.68	2		1	3025
SMM-1	60-69	SMM	male	White	IgG Kappa	2.35	20%	61.79	1	1476	0	2796
SMM-2	60-69	SMM	female	White	IgG Kappa	1.52	12.50%	3.29	2		0	4006
SMM-3	50-59	SMM	male	White	IgG Kappa	0.93	20%	3.69	2		0	3062
SMM-4	50-59	SMM	male	White	IgA Kappa	0.43	11%	1.52	2		1	3125
SMM-5	50-59	SMM	female	White	IgG Kappa	2.74	30%	4.68	2		1	2188
SMM-6	60-69	SMM	male	White	IgG Kappa	3.51	50%	16.49	1	321	1	1895
SMM-7	50-59	SMM	female	White	IgA Kappa	2.94	35%	11.94	1	2859	1	3868
SMM-8	50-59	SMM	male	White	IgA Kappa	0.31	10%	11.89	1	1393	1	1659
SMM-9	40-49	SMM	male	White	IgA Kappa	0.36	20%	3.03	2		1	1738
SMM-10	80-89	SMM	male	White	IgG Lambda	0.91	12.50%	52.66	1	2186	1	2446
SMM-11	60-69	SMM	male	White	IgG Kappa	1.69	35%	66.79	1	120	0	1565
SMM-12	70-79	SMM	female	White	IgG Kappa	0.69	75%	219.29	1	423	1	1921
MM-1	60-69	MM	male	White	IgA Lambda	0.45	100%	1,406.13	3	0	nan	2246
MM-2	50-59	MM	male	White	IgG Kappa	4.71	28%	80.64	3		nan	1783
MM-3	70-79	MM	male	White	IgG Kappa	2.64	25%	24.97	1	2591	nan	4912
MM-4	70-79	MM	male	White	Lambda light chain	0.23	40%	165.18	3	0	nan	1400
MM-5	60-69	MM	male	Hispanic	IgG Kappa	4.15	65%	2.44	3	0	nan	1748
MM-6	60-69	MM	female	White	IgG Kappa	4.35	40%	4.00	3	2	nan	1765
MM-7	50-59	MM	female	White	IgG Kappa	2.99	50%	1.62	3	426	nan	2130
MM-8	60-69	MM	male	White	IgG Kappa	3.06	25%	87.47	1	1476	nan	2796

Table A.3: Quality metrics for scRNAseq samples, including whether the sample was fresh or frozen, batch ID, n cells retained after removing low quality cells and non-CD138+ cells (QC), and median UMI post-QC.

sample_ID	sorting_tag	fresh_or_frozen	batch	ncells_persample	median_UMI
NBM-1	138P	frozen	batch 1	760	9020.5
NBM-2	138P	fresh	batch 2	857	13635
NBM-3	138P	frozen	batch 3	51	11979
NBM-4	138P	frozen	batch 4	160	14203
NBM-6	138P	frozen	batch 4	679	10909
NBM-7	138P	frozen	batch 4	1569	11542
NBM-8	138P	frozen	batch 5	1807	12053
NBM-10	138P	frozen	batch 5	2606	6280
NBM-11	138P	frozen	batch 5	840	9235
MGUS-1	138P	frozen	batch 6	133	9690
MGUS-2	138P	frozen	batch 4	136	7018
MGUS-3	138P	frozen	batch 3	371	14269
MGUS-4	138P	frozen	batch 6	62	9916.5
MGUS-5	138P	frozen	batch 6	53	6183
MGUS-6	138P	frozen	batch 3	82	22070.5
SMM-1	138P	frozen	batch 4	439	5076
SMM-2	138P	frozen	batch 4	1857	10147
SMM-3	138P	frozen	batch 3	349	27154
SMM-4	138P	frozen	batch 1	136	7250
SMM-5	138P	frozen	batch 3	40	9968
SMM-6	138P	frozen	batch 4	1140	8879
SMM-7	138P	frozen	batch 4	2049	12242
SMM-8	138P	frozen	batch 4	711	14195
SMM-9	138P	frozen	batch 4	1253	11935
SMM-10	138P	frozen	batch 3	67	20770
SMM-11	138P	frozen	batch 4	106	4196
SMM-12	138P	frozen	batch 3	284	6890
MM-1	138P	frozen	batch 4	2887	3553
MM-2	138P	fresh	batch 1	3414	7168
MM-3	138P	frozen	batch 2	950	23678.5
MM-4	138P	fresh	batch 2	591	21884
MM-5	138P	frozen	batch 4	1463	8415
MM-6	138P	fresh	batch 3	832	16172.5
MM-7	138P	frozen	batch 4	100	10287
MM-8	138P	frozen	batch 4	553	3178

Table A.4: **Full list of NMF signatures.** List of top genes and descriptions for all 28 signatures discovered using Bayesian NMF.

signature	description	type	top_genes
W1	unknown		['JUNB', 'ZFP36', 'NFKBIA', 'IER2']
W2	MM-1	patient specific	['MTDH', 'HLA-A', 'IFI27', 'SNHG25']
W3	t(11;14) associated		['CCND1', 'TSC22D3', 'RP5-887A10.1', 'RGS13']
W4	HLA class II		['HLA-DRA', 'HLA-DRB1', 'HLA-DPA1', 'HLA-DPB1']
W5	histones		['HIST1H1C', 'HIST1H2AC', 'HIST1H2BC', 'KIAA0556']
W6	unknown		['DUSP4', 'GADD45A', 'BTG2', 'LAMP5']
W7	MM-4	patient specific	['IFI6', 'PTP4A3', 'HLA-A', 'LAG3']
W8	t(14;20) associated		['ITGB7', 'AC233755.2', 'SPP1', 'CCND2']
W9	extracellular signaling		['LGALS1', 'VIM', 'ACTB', 'S100A6']
W10	RPL36A	single gene	['RPL36A', 'NBEAL1', 'IFITM1', 'LAMP5']
W11	proliferation		['HIST1H4C', 'STMN1', 'TUBA1B', 'HMGB2']
W12	monocytic ambient contamination	contamination	['SAT1', 'S100A9', 'S100A8', 'CRIP1']
W13	FOS	single gene	['FOS', 'RPPH1', 'ID1', 'CDKL3']
W14	unknown		['KLF6', 'TSC22D3', 'ANKRD28', 'KLF2']
W15	SMMh-5	patient specific	['AREG', 'KLF4', 'NEAT1', 'CTA-292E10.6']
W16	normal plasma cell signature		['CD27', 'CD79A', 'TXNIP', 'JSRP1']
W17	MM-5	patient specific	['EIF3L', 'MYC', 'RPL36A', 'JUNB']
W18	JUN	single gene	['JUN', 'FOSB', 'EGR1', 'IER2']
W19	SMMh-4	patient specific	['CST3', 'ITM2C', 'MTDH', 'TIMP1']
W20	protein synthesis (ser/thr kinase; elongation factors; pre/mRNA editing)		['HNRNPH1', 'PIM2', 'C16orf54', 'PHKG1']
W21	HSPA5	single gene	['HSPA5', 'CST6', 'EPCAM', 'HYOU1']
W22	TMSB4X	single gene	['TMSB4X', 'PMAIP1', 'LAPTM5', 'FNBP1']
W23	MM-3	patient specific	['FRZB', 'CCL3', 'CCL4', 'DKK1']
W24	interferon inducible		['ISG15', 'MX1', 'TNFSF10', 'LY6E']
W25	UBC	single gene	['UBC', 'C1GALT1C1', 'SLC3A2', 'LMF1']
W26	unknown		['HLA-A', 'ITM2C', 'PRR15', 'ACTB']
W27	nuclear genes (opposite expression pattern to MALAT1)		['NEAT1', 'DDX17', 'ANKRD12', 'FOXO3']
W28	CXCR4 & regulators		['CXCR4', 'RGS1', 'RGS2', 'ICA1L']

Appendix B

Supplementary Data for Chapter 5

B.1 Full results for all clinical prediction tasks

Table B.1: Full set of results for the triple HLCA task. Average across folds ± SEM are shown.

	AUC			Accuracy			Weighted F1		
model	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E
scSet	0.89±0.05	0.83±0.06	0.74±0.11	0.75±0.06	0.66±0.07	0.65±0.07	0.78±0.06	0.68±0.07	0.66±0.08
scSet w/o diffusion	0.73±0.07	0.71±0.05	0.72±0.07	0.61±0.05	0.45±0.07	0.51±0.08	0.61±0.06	0.47±0.08	0.53±0.09
scSet w/ flow decoder	0.73±0.07	0.66±0.06	0.73±0.07	0.58±0.06	0.49±0.09	0.55±0.07	0.57±0.07	0.43±0.09	0.57±0.07
abMIL w/ diffusion	0.71±0.07	0.7±0.05	0.71±0.05	0.65±0.08	0.53±0.06	0.48±0.06	0.62±0.08	0.57±0.06	0.52±0.05
abMIL w/o diffusion	0.71±0.07	0.68±0.05	0.73±0.05	0.58±0.06	0.39±0.06	0.49±0.06	0.57±0.07	0.38±0.07	0.52±0.06
average	0.71±0.06	0.62±0.05	0.64±0.05	0.62±0.07	0.44±0.05	0.46±0.04	0.58±0.07	0.49±0.06	0.5±0.05
cell type fractions	0.79±0.04	0.74±0.1	NaN	0.58±0.06	0.47±0.09	NaN	0.6±0.07	0.51±0.09	NaN
cell type means	0.9±0.03	0.84±0.05	NaN	0.7±0.04	0.63±0.04	NaN	0.73±0.04	0.68±0.04	NaN
cell type fracs+means	0.9±0.03	0.8±0.06	NaN	0.7±0.04	0.66±0.05	NaN	0.72±0.04	0.7±0.05	NaN
kmeans30	0.92±0.05	0.83±0.08	NaN	0.76±0.05	0.58±0.07	NaN	0.77±0.05	0.63±0.06	NaN
kmeans60	0.9±0.03	0.79±0.07	NaN	0.78±0.04	0.63±0.07	NaN	0.79±0.04	0.66±0.06	NaN

Table B.2: Full set of results for the SLE task. Average across folds ± SEM are shown.

	AUC			Accuracy			Weighted F1		
model	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E
scSet	0.98±0.01	0.98±0.01	0.99±0.0	0.93±0.02	0.94±0.01	0.95±0.01	0.93±0.02	0.94±0.01	0.95±0.01
scSet w/o diffusion	0.98±0.01	0.95±0.02	0.97±0.01	0.92±0.02	0.84±0.03	0.92±0.02	0.92±0.02	0.83±0.04	0.92±0.02
scSet w/ flow decoder	0.95±0.01	0.88±0.02	0.97±0.01	0.87±0.02	0.77±0.04	0.92±0.02	0.87±0.02	0.76±0.04	0.92±0.01
abMIL w/ diffusion	0.95±0.01	0.95±0.02	0.96±0.01	0.87±0.02	0.9±0.01	0.92±0.01	0.87±0.02	0.9±0.01	0.92±0.01
abMIL w/o diffusion	0.94±0.01	0.91±0.01	0.97±0.01	0.87±0.02	0.82±0.01	0.9±0.02	0.87±0.02	0.82±0.01	0.9±0.02
average	0.95±0.01	0.87±0.02	0.86±0.04	0.88±0.02	0.75±0.03	0.77±0.03	0.88±0.02	0.75±0.03	0.77±0.03
cell type fractions	0.93±0.01	0.79±0.04	NaN	0.83±0.02	0.73±0.02	NaN	0.83±0.02	0.72±0.03	NaN
cell type means	0.98±0.01	0.93±0.02	NaN	0.94±0.01	0.86±0.03	NaN	0.94±0.01	0.86±0.03	NaN
cell type fracs+means	0.99±0.0	0.96±0.01	NaN	0.95±0.01	0.9±0.02	NaN	0.95±0.01	0.9±0.02	NaN
kmeans30	0.98±0.01	0.97±0.01	NaN	0.94±0.02	0.92±0.02	NaN	0.94±0.02	0.92±0.02	NaN
kmeans60	0.98±0.01	0.98±0.01	NaN	0.89±0.02	0.92±0.01	NaN	0.89±0.02	0.92±0.01	NaN

Table B.3: Full set of results for the binary COVID task. Average across folds ± SEM are shown.

-1cm-1cm

	AUC			Accuracy			Weighted F1		
model	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E
scSet	0.98±0.01	0.98±0.02	0.98±0.01	0.95±0.02	0.93±0.02	0.92±0.02	0.95±0.02	0.93±0.02	0.93±0.02
scSet w/o diffusion	0.95±0.02	0.96±0.02	0.82±0.1	0.9±0.02	0.86±0.02	0.88±0.03	0.9±0.03	0.86±0.02	0.88±0.03
scSet w/ flow decoder	0.93±0.02	0.79±0.04	0.94±0.05	0.89±0.03	0.75±0.02	0.86±0.03	0.9±0.03	0.71±0.04	0.87±0.03
abMIL w/ diffusion	0.92±0.03	0.86±0.06	0.88±0.06	0.88±0.03	0.83±0.03	0.85±0.03	0.87±0.03	0.83±0.03	0.85±0.03
abMIL w/o diffusion	0.93±0.03	0.92±0.03	0.9±0.04	0.89±0.03	0.85±0.03	0.86±0.02	0.88±0.03	0.86±0.03	0.84±0.03
average	0.94±0.02	0.85±0.03	0.86±0.04	0.88±0.02	0.79±0.04	0.79±0.05	0.88±0.02	0.8±0.04	0.81±0.04
cell type fractions	0.89±0.04	0.61±0.06	NaN	0.85±0.02	0.67±0.04	NaN	0.84±0.02	0.68±0.04	NaN
cell type means	0.96±0.02	0.97±0.02	NaN	0.88±0.02	0.93±0.03	NaN	0.87±0.03	0.92±0.03	NaN
cell type fracs+means	0.96±0.02	0.97±0.02	NaN	0.88±0.02	0.91±0.03	NaN	0.87±0.03	0.92±0.03	NaN
kmeans30	0.95±0.02	0.9±0.04	NaN	0.92±0.02	0.87±0.04	NaN	0.92±0.03	0.86±0.04	NaN
kmeans60	0.96±0.01	0.97±0.02	NaN	0.94±0.02	0.9±0.02	NaN	0.94±0.02	0.9±0.02	NaN

Table B.4: Full set of results for the binary HLCA task. Average across folds ± SEM are shown.

	AUC			Accuracy			Weighted F1		
model	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E	linear probe	MLP	FT-E2E
scSet	0.78±0.06	0.81±0.06	0.85±0.03	0.78±0.06	0.87±0.04	0.83±0.06	0.76±0.09	0.87±0.04	0.83±0.06
scSet w/o diffusion	0.64±0.09	0.58±0.1	0.71±0.08	0.8±0.07	0.71±0.07	0.81±0.06	0.77±0.09	0.73±0.06	0.82±0.06
scSet w/ flow decoder	0.47±0.09	0.48±0.15	0.73±0.04	0.81±0.07	0.72±0.1	0.84±0.04	0.75±0.09	0.67±0.11	0.85±0.04
abMIL w/ diffusion	0.59±0.06	0.66±0.08	0.68±0.06	0.79±0.07	0.77±0.07	0.79±0.05	0.74±0.09	0.78±0.07	0.78±0.06
abMIL w/o diffusion	0.46±0.13	0.58±0.11	0.45±0.09	0.81±0.07	0.7±0.07	0.7±0.07	0.75±0.09	0.7±0.08	0.7±0.07
average	0.51±0.12	0.64±0.12	0.48±0.06	0.81±0.07	0.69±0.06	0.63±0.04	0.75±0.09	0.73±0.06	0.67±0.05
cell type fractions	0.81±0.07	0.53±0.08	NaN	0.79±0.07	0.7±0.06	NaN	0.77±0.09	0.73±0.07	NaN
cell type means	0.9±0.07	0.82±0.06	NaN	0.86±0.04	0.72±0.07	NaN	0.85±0.05	0.76±0.06	NaN
cell type fracs+means	0.9±0.06	0.81±0.08	NaN	0.86±0.04	0.79±0.04	NaN	0.85±0.05	0.82±0.04	NaN
kmeans30	0.9±0.04	0.82±0.04	NaN	0.88±0.03	0.87±0.04	NaN	0.89±0.03	0.87±0.04	NaN
kmeans60	0.92±0.04	0.86±0.03	NaN	0.87±0.04	0.82±0.03	NaN	0.86±0.04	0.83±0.04	NaN

References

[1] Z. Liu, Z. Yang, J. Wu, W. Zhang, Y. Sun, C. Zhang, G. Bai, L. Yang, H. Fan, Y. Chen, et al. "A single-cell atlas reveals immune heterogeneity in anti-PD-1-treated non-small cell lung cancer". In: *Cell* (2025).

[2] R. A. Kyle and S. V. Rajkumar. "Multiple myeloma". In: *Blood* 111.6 (Mar. 2008), pp. 2962–2972.

[3] O. Landgren. "Monoclonal gammopathy of undetermined significance and smoldering multiple myeloma: biological insights and early treatment strategies". In: *Hematology* 2013.1 (Dec. 2013), pp. 478–487.

[4] S. V. Rajkumar. "Multiple myeloma: 2011 update on diagnosis, risk-stratification, and management". In: *Am. J. Hematol.* 86.1 (2011), pp. 57–65.

[5] S. Manier, K. Z. Salem, J. Park, D. A. Landau, G. Getz, and I. M. Ghobrial. "Genomic complexity of multiple myeloma and its clinical implications". In: *Nat. Rev. Clin. Oncol.* 14.2 (2017), p. 100.

[6] V. Y. F. Tan and C. Févotte. "Automatic Relevance Determination in Nonnegative Matrix Factorization with the β-Divergence". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 35.7 (2013), pp. 1592–1605.

[7] F. Yang, W. Wang, F. Wang, Y. Fang, D. Tang, J. Huang, H. Lu, and J. Yao. "scBERT as a large-scale pretrained deep language model for cell type annotation of single-cell RNA-seq data". en. In: *Nature Machine Intelligence* 4.10 (Sept. 2022), pp. 852–866.

[8] H. Cui, C. Wang, H. Maan, K. Pang, F. Luo, N. Duan, and B. Wang. "scGPT: toward building a foundation model for single-cell multi-omics using generative AI". In: *Nature Methods* (2024), pp. 1–11.

[9] Y. Rosen, Y. Roohani, A. Agarwal, L. Samotorčan, T. S. Consortium, S. R. Quake, and J. Leskovec. "Universal cell embeddings: A foundation model for cell biology". In: *bioRxiv* (2023), pp. 2023–11.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by J. Burstein, C. Doran, and T. Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[11] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. "Improving language understanding by generative pre-training". In: (2018).

[12] R. Lopez, J. Regier, M. B. Cole, M. I. Jordan, and N. Yosef. "Deep generative modeling for single-cell transcriptomics". In: *Nature methods* 15.12 (2018), pp. 1053–1058.

[13] A. 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 (2021), e2016239118. DOI: 10.1073/pnas.2016239118. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.2016239118. URL: https://www.pnas.org/doi/abs/10.1073/pnas.2016239118.

[14] A. Radford et al. "Learning Transferable Visual Models From Natural Language Supervision". In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by M. Meila and T. Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 8748–8763.

[15] X. Yang et al. "A large language model for electronic health records". en. In: *NPJ Digit Med* 5.1 (Dec. 2022), p. 194.

[16] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". In: *Journal of Machine Learning Research* 21.140 (2020), pp. 1–67. URL: http://jmlr.org/papers/v21/20-074.html.

[17] T. Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[18] R. A. Kyle, E. D. Remstein, T. M. Therneau, A. Dispenzieri, P. J. Kurtin, J. M. Hodnefield, D. R. Larson, M. F. Plevak, D. F. Jelinek, R. Fonseca, et al. "Clinical course and prognosis of smoldering (asymptomatic) multiple myeloma". In: *N. Engl. J. Med.* 356.25 (2007), pp. 2582–2590.

[19] R. A. Kyle, D. R. Larson, T. M. Therneau, A. Dispenzieri, S. Kumar, J. R. Cerhan, and S. V. Rajkumar. "Long-term follow-up of monoclonal gammopathy of undetermined significance". In: *N. Engl. J. Med.* 378.3 (2018), pp. 241–249.

[20] A. Lakshman, S. V. Rajkumar, F. K. Buadi, M. Binder, M. A. Gertz, M. Q. Lacy, A. Dispenzieri, D. Dingli, A. L. Fonder, S. R. Hayman, et al. "Risk stratification of smoldering multiple myeloma incorporating revised IMWG diagnostic criteria". In: *Blood Cancer J.* 8.6 (2018), pp. 1–10.

[21] S. V. Rajkumar, V. Gupta, R. Fonseca, A. Dispenzieri, W. I. Gonsalves, D. Larson, R. P. Ketterling, J. A. Lust, R. A. Kyle, and S. K. Kumar. "Impact of primary molecular cytogenetic abnormalities and risk of progression in smoldering multiple myeloma". In: *Leukemia* 27.8 (2013), pp. 1738–1744.

[22] N. Bolli et al. "Genomic patterns of progression in smoldering multiple myeloma". en. In: *Nat. Commun.* 9.1 (Aug. 2018), p. 3363.

[23] B. Oben et al. "Whole-genome sequencing reveals progressive versus stable myeloma precursor conditions as two distinct entities". en. In: *Nat. Commun.* 12.1 (Mar. 2021), p. 1861.

[24] M.-V. Mateos et al. "International Myeloma Working Group risk stratification model for smoldering multiple myeloma (SMM)". en. In: *Blood Cancer J.* 10.10 (Oct. 2020), p. 102.

[25] M. Bustoros et al. "Genomic Profiling of Smoldering Multiple Myeloma Identifies Patients at a High Risk of Disease Progression". en. In: *J. Clin. Oncol.* 38.21 (July 2020), pp. 2380–2389.

[26] K. Misund et al. "MYC dysregulation in the progression of multiple myeloma". en. In: *Leukemia* 34.1 (Jan. 2020), pp. 322–326.

[27] W. J. Chng, G. F. Huang, T. H. Chung, S. B. Ng, N. Gonzalez-Paz, T. Troska-Price, G. Mulligan, M. Chesi, P. L. Bergsagel, and R. Fonseca. "Clinical and biological implications of MYC activation: a common difference between MGUS and newly diagnosed multiple myeloma". In: *Leukemia* 25.6 (2011), pp. 1026–1035.

[28] G. Ledergor, A. Weiner, M. Zada, S.-Y. Wang, Y. C. Cohen, M. E. Gatt, N. Snir, H. Magen, M. Koren-Michowitz, K. Herzog-Tzarfati, et al. "Single cell dissection of plasma cell heterogeneity in symptomatic and asymptomatic myeloma". In: *Nat. Med.* 24.12 (2018), pp. 1867–1876.

[29] J. S. Jang, Y. Li, A. K. Mitra, L. Bi, A. Abyzov, A. J. van Wijnen, L. B. Baughn, B. Van Ness, V. Rajkumar, S. Kumar, et al. "Molecular signatures of multiple myeloma progression through single cell RNA-Seq". In: *Blood Cancer J.* 9.1 (2019), pp. 1–10.

[30] R. Boiarsky, N. J. Haradhvala, J.-B. Alberge, R. Sklavenitis-Pistofidis, T. H. Mouhieddine, O. Zavidij, M.-C. Shih, D. Firer, M. Miller, H. El-Khoury, et al. "Single cell characterization of myeloma and its precursor conditions reveals transcriptional signatures of early tumorigenesis". In: *Nature Communications* 13.1 (2022), p. 7040.

[31] O. Zavidij, N. J. Haradhvala, T. H. Mouhieddine, R. Sklavenitis-Pistofidis, S. Cai, M. Reidy, M. Rahmat, A. Flaifel, B. Ferland, N. K. Su, et al. "Single-cell RNA sequencing reveals compromised immune microenvironment in precursor stages of multiple myeloma". In: *Nature Cancer* 1.5 (2020), pp. 493–506.

[32] V. A. Traag, L. Waltman, and N. J. Van Eck. "From Louvain to Leiden: guaranteeing well-connected communities". In: *Sci. Rep.* 9.1 (2019), pp. 1–12.

[33] N. Robillard, H. Avet-Loiseau, R. Garand, P. Moreau, D. Pineau, M.-J. Rapp, J.-L. Harousseau, and R. Bataille. "CD20 is associated with a small mature plasma cell morphology and t (11; 14) in multiple myeloma". In: *Blood* 102.3 (2003), pp. 1070–1071.

[34] G. Mateo, M. Castellanos, A. Rasillo, N. C. Gutiérrez, M. A. Montalbán, M. L. Martín, J. M. Hernández, M. C. López-Berges, L. Montejano, J. Bladé, et al. "Genetic abnormalities and patterns of antigenic expression in multiple myeloma". In: *Clin. Cancer Res.* 11.10 (2005), pp. 3661–3667.

[35] M. E. Ritchie, B. Phipson, D. I. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth. "limma powers differential expression analyses for RNA-sequencing and microarray studies". In: *Nucleic Acids Res.* 43.7 (2015), e47–e47.

[36] C. W. Law, Y. Chen, W. Shi, and G. K. Smyth. "voom: Precision weights unlock linear model analysis tools for RNA-seq read counts". In: *Genome Biol.* 15.2 (2014), pp. 1–17.

[37] P. Moreau, N. Robillard, G. Jégo, C. Pellat, S. L. Gouill, S. Thoumi, H. Avet-Loiseau, J.-L. Harousseau, and R. Bataille. "Lack of CD27 in myeloma delineates different presentation and outcome". In: *Br. J. Haematol.* 132.2 (2006), pp. 168–170.

[38] J. E. J. Guikema, S. Hovenga, E. Vellenga, J. J. Conradie, W. H. Abdulahad, R. Bekkema, J. W. Smit, F. Zhan, J. Shaughnessy Jr, and N. A. Bos. "CD27 is heterogeneously expressed in multiple myeloma: low CD27 expression in patients with high-risk disease". In: *Br. J. Haematol.* 121.1 (2003), pp. 36–43.

[39] T. K. Morgan, S. Zhao, K. L. Chang, T. L. Haddix, E. Domanay, P. J. Cornbleet, D. A. Arber, and Y. Natkunam. "Low CD27 expression in plasma cell dyscrasias correlates with high-risk disease: an immunohistochemical analysis". In: *Am. J. Clin. Pathol.* 126.4 (2006), pp. 545–551.

[40] F. E. Davies, A. M. Dring, C. Li, A. C. Rawstron, M. A. Shammas, S. M. O'Connor, J. A. L. Fenton, T. Hideshima, D. Chauhan, I. T. Tai, et al. "Insights into the multistep transformation of MGUS to myeloma using microarray expression analysis". In: *Blood* 102.13 (2003), pp. 4504–4511.

[41] J.-Y. Wang et al. "STIM1 overexpression promotes colorectal cancer progression, cell motility and COX-2 expression". en. In: *Oncogene* 34.33 (Aug. 2015), pp. 4358–4367.

[42] M. Debant et al. "STIM1 at the plasma membrane as a new target in progressive chronic lymphocytic leukemia". en. In: *J Immunother Cancer* 7.1 (Apr. 2019), p. 111.

[43] W. Wang, Y. Ren, L. Wang, W. Zhao, X. Dong, J. Pan, H. Gao, and Y. Tian. "Orai1 and Stim1 Mediate the Majority of Store-Operated Calcium Entry in Multiple Myeloma and Have Strong Implications for Adverse Prognosis". en. In: *Cell. Physiol. Biochem.* 48.6 (Aug. 2018), pp. 2273–2285.

[44] A. L. Garfall, M. V. Maus, W.-T. Hwang, S. F. Lacey, Y. D. Mahnke, J. J. Melenhorst, Z. Zheng, D. T. Vogl, A. D. Cohen, B. M. Weiss, et al. "Chimeric antigen receptor T cells against CD19 for multiple myeloma". In: *N. Engl. J. Med.* 373.11 (2015), pp. 1040–1047.

[45] T. Nerreter, S. Letschert, R. Götz, S. Doose, S. Danhof, H. Einsele, M. Sauer, and M. Hudecek. "Super-resolution microscopy reveals ultra-low CD19 expression on myeloma cells that triggers elimination by CD19 CAR-T". In: *Nat. Commun.* 10.1 (2019), pp. 1–11.

[46] B. Paiva et al. "Clinical significance of CD81 expression by clonal plasma cells in high-risk smoldering and symptomatic multiple myeloma patients". en. In: *Leukemia* 26.8 (Aug. 2012), pp. 1862–1869.

[47] P. R. Tembhare, C. Yuan, N. Korde, I. Maric, K. Calvo, M. A. Yancey, M. Mulquin, O. Landgren, and M. Stetler-Stevenson. "CD81: A Novel, Specific and Highly Sensitive Marker in Flow Cytometric Diagnosis of Plasma Cell Dyscrasia". In: *Blood* 118.21 (Nov. 2011), p. 2880.

[48] J. J. Li and D. Xie. "RACK1, a versatile hub in cancer". In: *Oncogene* 34.15 (2015), pp. 1890–1898.

[49] L. Zhang, Y. Xu, L. Wang, and H. Liu. "Role of RACK1 on cell proliferation, adhesion, and bortezomib-induced apoptosis in multiple myeloma". In: *Int. J. Biol. Macromol.* 121 (2019), pp. 1077–1085.

[50] M. Sarıman, N. Abacı, S. S. Ekmekçi, A. Çakiris, F. P. Paçal, D. Üstek, M. Ayer, M. N. Yenerel, S. Beşışık, K. Çefle, et al. "Investigation of gene expressions of myeloma cells in the bone marrow of multiple myeloma patients by transcriptome analysis". In: *Balkan Med. J.* 36.1 (2019), p. 23.

[51] S. Trezise, A. Karnowski, P. L. Fedele, S. Mithraprabhu, Y. Liao, K. D'costa, A. J. Kueh, M. P. Hardy, C. M. Owczarek, M. J. Herold, et al. "Mining the plasma cell transcriptome for novel cell surface proteins". In: *Int. J. Mol. Sci.* 19.8 (2018), p. 2161.

[52] Z. Zeng, J. Lin, K. Zhang, X. Guo, X. Zheng, A. Yang, and J. Chen. "Single cell RNA-seq data and bulk gene profiles reveal a novel signature of disease progression in multiple myeloma". en. In: *Cancer Cell Int.* 21.1 (Sept. 2021), p. 511.

[53] A. Taylor-Weiner, F. Aguet, N. J. Haradhvala, et al. "Scaling computational genomics to millions of individuals with GPUs". In: *Genome Biol.* 20.228 (2019).

[54] F. Zhan, Y. Huang, S. Colla, J. P. Stewart, I. Hanamura, S. Gupta, J. Epstein, S. Yaccoby, J. Sawyer, B. Burington, et al. "The molecular classification of multiple myeloma". In: *Blood* 108.6 (2006), pp. 2020–2028.

[55] A. Broyl, D. Hose, H. Lokhorst, Y. de Knegt, J. Peeters, A. Jauch, U. Bertsch, A. Buijs, M. Stevens-Kroef, H. B. Beverloo, et al. "Gene expression profiling for molecular classification of multiple myeloma in newly diagnosed patients". In: *Blood, The Journal of the American Society of Hematology* 116.14 (2010), pp. 2543–2553.

[56] T. R. Ullah. "The role of CXCR4 in multiple myeloma: Cells' journey from bone marrow to beyond". In: *Journal of bone oncology* 17 (2019), p. 100253.

[57] S. J. Coniglio. "Role of tumor-derived chemokines in osteolytic bone metastasis". In: *Front. Endocrinol.* 9 (2018), p. 313.

[58] H.-K. Pak, M. Gil, Y. Lee, H. Lee, A.-N. Lee, J. Roh, and C.-S. Park. "Regulator of G protein signaling 1 suppresses CXCL12-mediated migration and AKT activation in RPMI 8226 human plasmacytoma cells and plasmablasts". en. In: *PLoS One* 10.4 (Apr. 2015), e0124793.

[59] A. Liberzon, C. Birger, H. Thorvaldsdóttir, M. Ghandi, J. P. Mesirov, and P. Tamayo. "The molecular signatures database hallmark gene set collection". In: *Cell systems* 1.6 (2015), pp. 417–425.

[60] O. Zavidij, N. J. Haradhvala, T. H. Mouhieddine, R. Sklavenitis-Pistofidis, S. Cai, M. Reidy, M. Rahmat, A. Flaifel, B. Ferland, N. K. Su, et al. "Single-cell RNA sequencing reveals compromised immune microenvironment in precursor stages of multiple myeloma". In: *Nature cancer* 1.5 (2020), pp. 493–506.

[61] A. K. Dutta, J. L. Fink, J. P. Grady, G. J. Morgan, C. G. Mullighan, L. B. To, D. R. Hewett, and A. C. W. Zannettino. "Subclonal evolution in disease progression from MGUS/SMM to multiple myeloma is characterised by clonal stability". en. In: *Leukemia* 33.2 (Feb. 2019), pp. 457–468.

[62] A. K. Dutta, J.-B. Alberge, R. Sklavenitis-Pistofidis, E. D. Lightbody, G. Getz, and I. M. Ghobrial. "Single-cell profiling of tumour evolution in multiple myeloma - opportunities for precision medicine". en. In: *Nat. Rev. Clin. Oncol.* 19.4 (Apr. 2022), pp. 223–236.

[63] C. Y. Lin, J. Lovén, P. B. Rahl, R. M. Paranal, C. B. Burge, J. E. Bradner, T. I. Lee, and R. A. Young. "Transcriptional amplification in tumor cells with elevated c-Myc". In: *Cell* 151.1 (2012), pp. 56–67.

[64] E. Tian, F. Zhan, R. Walker, E. Rasmussen, Y. Ma, B. Barlogie, and J. D. Shaughnessy Jr. "The role of the Wnt-signaling antagonist DKK1 in the development of osteolytic lesions in multiple myeloma". In: *N. Engl. J. Med.* 349.26 (2003), pp. 2483–2494.

[65] H. van Andel, K. A. Kocemba, M. Spaargaren, and S. T. Pals. "Aberrant Wnt signaling in multiple myeloma: molecular mechanisms and targeting options". In: *Leukemia* 33.5 (2019), pp. 1063–1075.

[66] D. Kim, C. Y. Park, B. C. Medeiros, and I. L. Weissman. "CD19- CD45 low/- CD38 high/CD138+ plasma cells enrich for human tumorigenic myeloma cells". In: *Leukemia* 26.12 (2012), pp. 2530–2537.

[67] J. Shi, G. J. Tricot, T. K. Garg, P. A. Malaviarachchi, S. M. Szmania, R. E. Kellum, B. Storrie, A. Mulder, J. D. Shaughnessy Jr, B. Barlogie, et al. "Bortezomib down-regulates the cell-surface expression of HLA class I and enhances natural killer cell–mediated lysis of myeloma". In: *Blood, The Journal of the American Society of Hematology* 111.3 (2008), pp. 1309–1317.

[68] I. S. Nijhof, R. W. J. Groen, H. M. Lokhorst, B. Van Kessel, A. C. Bloem, J. Van Velzen, R. de Jong-Korlaar, H. Yuan, W. A. Noort, S. K. Klein, et al. "Upregulation of CD38 expression on multiple myeloma cells by all-trans retinoic acid improves the efficacy of daratumumab". In: *Leukemia* 29.10 (2015), pp. 2039–2049.

[69] S. P. Treon, C. Mitsiades, N. Mitsiades, G. Young, D. Doss, R. Schlossman, and K. C. Anderson. "Tumor cell expression of CD59 is associated with resistance to CD20 serotherapy in patients with B-cell malignancies". In: *J. Immunother.* 24.3 (2001), pp. 263–271.

[70] N. Hosen, H. Ichihara, A. Mugitani, Y. Aoyama, Y. Fukuda, S. Kishida, Y. Matsuoka, H. Nakajima, M. Kawakami, T. Yamagami, et al. "CD48 as a novel molecular target for antibody therapy in multiple myeloma". In: *Br. J. Haematol.* 156.2 (2012), pp. 213–224.

[71] Y. Kawano, O. Zavidij, J. Park, M. Moschetta, K. Kokubun, T. H. Mouhieddine, S. Manier, Y. Mishima, N. Murakami, M. Bustoros, et al. "Blocking IFNAR1 inhibits multiple myeloma–driven Treg expansion and immunosuppression". In: *J. Clin. Invest.* 128.6 (2018), pp. 2487–2499.

[72] F. A. Wolf, P. Angerer, and F. J. Theis. "SCANPY: large-scale single-cell gene expression data analysis". In: *Genome biology* 19 (2018), pp. 1–5.

[73] S. V. Rajkumar et al. "International Myeloma Working Group updated criteria for the diagnosis of multiple myeloma". en. In: *Lancet Oncol.* 15.12 (Nov. 2014), e538–48.

[74] B. Phipson, S. Lee, I. J. Majewski, W. S. Alexander, and G. K. Smyth. "Robust hyperparameter estimation protects against hypervariable genes and improves power to detect differential expression". In: *Ann. Appl. Stat.* 10.2 (2016), p. 946.

[75] D. A. Barbie, P. Tamayo, J. S. Boehm, S. Y. Kim, S. E. Moody, I. F. Dunn, A. C. Schinzel, P. Sandy, E. Meylan, C. Scholl, et al. "Systematic RNA interference reveals that oncogenic KRAS-driven cancers require TBK1". In: *Nature* 462.7269 (2009), pp. 108–112.

[76] M. Reich, T. Liefeld, J. Gould, J. Lerner, P. Tamayo, and J. P. Mesirov. "GenePattern 2.0". In: *Nat. Genet.* 38.5 (2006), pp. 500–501.

[77] L. B. Alexandrov, J. Kim, N. J. Haradhvala, M. N. Huang, A. W. Tian Ng, Y. Wu, A. Boot, K. R. Covington, D. A. Gordenin, E. N. Bergstrom, et al. "The repertoire of mutational signatures in human cancer". In: *Nature* 578.7793 (2020), pp. 94–101.

[78] A. Butler, P. Hoffman, P. Smibert, E. Papalexi, and R. Satija. "Integrating single-cell transcriptomic data across different conditions, technologies, and species". In: *Nature biotechnology* 36.5 (2018), pp. 411–420.

[79] J. Fan, K. Slowikowski, and F. Zhang. "Single-cell transcriptomics in cancer: computational challenges and opportunities". In: *Experimental & Molecular Medicine* 52.9 (2020), pp. 1452–1465.

[80] I. Tirosh, B. Izar, S. M. Prakadan, M. H. Wadsworth, D. Treacy, J. J. Trombetta, A. Rotem, C. Rodman, C. Lian, G. Murphy, et al. "Dissecting the multicellular ecosystem of metastatic melanoma by single-cell RNA-seq". In: *Science* 352.6282 (2016), pp. 189–196.

[81] A. Gavish, M. Tyler, A. C. Greenwald, R. Hoefflin, D. Simkin, R. Tschernichovsky, N. Galili Darnell, E. Somech, C. Barbolin, T. Antman, et al. "Hallmarks of transcriptional intratumour heterogeneity across a thousand tumours". In: *Nature* (2023), pp. 1–9.

[82] R. Beroukhim, C. H. Mermel, D. Porter, G. Wei, S. Raychaudhuri, J. Donovan, J. Barretina, J. S. Boehm, J. Dobson, M. Urashima, et al. "The landscape of somatic copy-number alteration across human cancers". In: *Nature* 463.7283 (2010), pp. 899–905.

[83] P. J. Hastings, J. R. Lupski, S. M. Rosenberg, and G. Ira. "Mechanisms of change in gene copy number". In: *Nature Reviews Genetics* 10.8 (2009), pp. 551–564.

[84] T. I. Zack, S. E. Schumacher, S. L. Carter, A. D. Cherniack, G. Saksena, B. Tabak, M. S. Lawrence, C.-Z. Zhang, J. Wala, C. H. Mermel, et al. "Pan-cancer patterns of somatic copy number alteration". In: *Nature genetics* 45.10 (2013), pp. 1134–1140.

[85] S. L. Carter, K. Cibulskis, E. Helman, A. McKenna, H. Shen, T. Zack, P. W. Laird, R. C. Onofrio, W. Winckler, B. A. Weir, et al. "Absolute quantification of somatic DNA alterations in human cancer". In: *Nature biotechnology* 30.5 (2012), pp. 413–421.

[86] X. Shao, N. Lv, J. Liao, J. Long, R. Xue, N. Ai, D. Xu, and X. Fan. "Copy number variation is highly correlated with differential gene expression: a pan-cancer study". In: *BMC medical genetics* 20 (2019), pp. 1–14.

[87] A. P. Patel, I. Tirosh, J. J. Trombetta, A. K. Shalek, S. M. Gillespie, H. Wakimoto, D. P. Cahill, B. V. Nahed, W. T. Curry, R. L. Martuza, et al. "Single-cell RNA-seq highlights intratumoral heterogeneity in primary glioblastoma". In: *Science* 344.6190 (2014), pp. 1396–1401.

[88] A. Bhattacharya, R. D. Bense, C. G. Urzúa-Traslaviña, E. G. de Vries, M. A. van Vugt, and R. S. Fehrmann. "Transcriptional effects of copy number alterations in a large set of human cancers". In: *Nature communications* 11.1 (2020), p. 715.

[89] S. C. Hicks, F. W. Townes, M. Teng, and R. A. Irizarry. "Missing data and technical variability in single-cell RNA-sequencing experiments". In: *Biostatistics* 19.4 (2018), pp. 562–578.

[90] T. Stuart, A. Butler, P. Hoffman, C. Hafemeister, E. Papalexi, W. M. Mauck, Y. Hao, M. Stoeckius, P. Smibert, and R. Satija. "Comprehensive integration of single-cell data". In: *Cell* 177.7 (2019), pp. 1888–1902.

[91] Y. Hao, S. Hao, E. Andersen-Nissen, W. M. Mauck, S. Zheng, A. Butler, M. J. Lee, A. J. Wilk, C. Darby, M. Zager, et al. "Integrated analysis of multimodal single-cell data". In: *Cell* 184.13 (2021), pp. 3573–3587.

[92] I. Korsunsky, N. Millard, J. Fan, K. Slowikowski, F. Zhang, K. Wei, Y. Baglaenko, M. Brenner, P.-r. Loh, and S. Raychaudhuri. "Fast, sensitive and accurate integration of single-cell data with Harmony". In: *Nature methods* 16.12 (2019), pp. 1289–1296.

[93] C. Xu, R. Lopez, E. Mehlman, J. Regier, M. I. Jordan, and N. Yosef. "Probabilistic harmonization and annotation of single-cell transcriptomics data with deep generative models". In: *Molecular systems biology* 17.1 (2021), e9620.

[94] J. Ding and A. Regev. "Deep generative model embedding of single-cell RNA-Seq profiles on hyperspheres and hyperbolic spaces". In: *Nature communications* 12.1 (2021), p. 2554.

[95] J. D. Welch, V. Kozareva, A. Ferreira, C. Vanderburg, C. Martin, and E. Z. Macosko. "Single-cell multi-omic integration compares and contrasts features of brain cell identity". In: *Cell* 177.7 (2019), pp. 1873–1887.

[96] L. Jerby-Arnon and A. Regev. "DIALOGUE maps multicellular programs in tissue from single-cell or spatial transcriptomics data". In: *Nature biotechnology* 40.10 (2022), pp. 1467–1477.

[97] T. Ashuach, M. I. Gabitto, R. V. Koodli, G.-A. Saldi, M. I. Jordan, and N. Yosef. "MultiVI: deep generative model for the integration of multimodal data". In: *Nature Methods* 20.8 (2023), pp. 1222–1231.

[98] E. Weinberger, R. Lopez, J.-C. Hütter, and A. Regev. "Disentangling shared and group-specific variations in single-cell transcriptomics data with multiGroupVI". In: *Machine Learning in Computational Biology*. PMLR. 2022, pp. 16–32.

[99] P. Boyeau, J. Hong, A. Gayoso, M. Jordan, E. Azizi, and N. Yosef. "Deep generative modeling for quantifying sample-level heterogeneity in single-cell omics". In: *bioRxiv* (2022), pp. 2022–10.

[100] L. Jerby-Arnon, C. Neftel, M. E. Shore, H. R. Weisman, N. D. Mathewson, M. J. McBride, B. Haas, B. Izar, A. Volorio, G. Boulay, et al. "Opposing immune and genetic mechanisms shape oncogenic programs in synovial sarcoma". In: *Nature medicine* 27.2 (2021), pp. 289–300.

[101] H.-O. Lee, Y. Hong, H. E. Etlioglu, Y. B. Cho, V. Pomella, B. Van den Bosch, J. Vanhecke, S. Verbandt, H. Hong, J.-W. Min, et al. "Lineage-dependent gene expression programs influence the immune landscape of colorectal cancer". In: *Nature genetics* 52.6 (2020), pp. 594–603.

[102] inferCNV of the Trinity CTAT Project. *inferCNV*. https://github.com/broadinstitute/inferCNV. Accessed: 2023-08-30.

[103] J. A. Castro-Mondragon, R. Riudavets-Puig, I. Rauluseviciute, R. Berhanu Lemma, L. Turchi, R. Blanc-Mathieu, J. Lucas, P. Boddie, A. Khan, N. Manosalva Pérez, et al. "JASPAR 2022: the 9th release of the open-access database of transcription factor binding profiles". In: *Nucleic acids research* 50.D1 (2022), pp. D165–D173.

[104] A. Gayoso, R. Lopez, G. Xing, P. Boyeau, V. Valiollah Pour Amiri, J. Hong, K. Wu, M. Jayasuriya, E. Mehlman, M. Langevin, et al. "A Python library for probabilistic analysis of single-cell omics data". In: *Nature biotechnology* 40.2 (2022), pp. 163–166.

[105] R. C. Geary. "The contiguity ratio and statistical mapping". In: *The incorporated statistician* 5.3 (1954), pp. 115–146.

[106] D. DeTomaso, M. G. Jones, M. Subramaniam, T. Ashuach, C. J. Ye, and N. Yosef. "Functional interpretation of single cell similarity maps". In: *Nature communications* 10.1 (2019), p. 4376.

[107] M. P. Gold, W. Ong, A. M. Masteller, J. A. Galindo, N. R. Park, R. A. Saurez, M. C. Vladoiu, L. K. Donovan, A. D. Walker, J. Benetatos, et al. "Developmental Basis of SHH Medulloblastoma Heterogeneity". In: *bioRxiv* (2022), pp. 2022–09.

[108] H. Cui, C. Wang, H. Maan, and B. Wang. "scGPT: Towards Building a Foundation Model for Single-Cell Multi-omics Using Generative AI". en. May 2023.

[109] C. V. Theodoris et al. "Transfer learning enables predictions in network biology". en. In: *Nature* 618.7965 (June 2023), pp. 616–624.

[110] M. Hao, J. Gong, X. Zeng, C. Liu, Y. Guo, X. Cheng, T. Wang, J. Ma, X. Zhang, and L. Song. "Large-scale foundation model on single-cell transcriptomics". In: *Nature Methods* 21.8 (2024), pp. 1481–1491.

[111] G. Heimberg et al. "A cell atlas foundation model for scalable search of similar human cells". In: *Nature* (2024).

[112] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[113] J. Du, P. Jia, Y. Dai, C. Tao, Z. Zhao, and D. Zhi. "Gene2vec: distributed representation of genes based on co-expression". en. In: *BMC Genomics* 20.Suppl 1 (Feb. 2019), p. 82.

[114] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al. "Rethinking attention with performers". In: *arXiv preprint arXiv:2009.14794* (2020).

[115] T. Dao, D. Fu, S. Ermon, A. Rudra, and C. Ré. "Flashattention: Fast and memory-efficient exact attention with io-awareness". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 16344–16359.

[116] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. Springer, 2009.

[117] R. Tibshirani. "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), pp. 267–288. ISSN: 00359246. URL: http://www.jstor.org/stable/2346178 (visited on 11/21/2024).

[118] Stanford University. Department of Statistics and R. Tibshirani. *Regression Shrinkage and Selection Via the Lasso*. en. 1994.

[119] G. X. Y. Zheng et al. "Massively parallel digital transcriptional profiling of single cells". en. In: *Nat. Commun.* 8 (Jan. 2017), p. 14049.

[120] L. Schirmer, D. Velmeshev, S. Holmqvist, M. Kaufmann, S. Werneburg, D. Jung, S. Vistnes, J. H. Stockley, A. Young, M. Steindel, et al. "Neuronal vulnerability and multilineage diversity in multiple sclerosis". In: *Nature* 573.7772 (2019), pp. 75–82.

[121] J. Chen, H. Xu, W. Tao, Z. Chen, Y. Zhao, and J.-D. J. Han. "Transformer for one stop interpretable cell type annotation". In: *Nature Communications* 14.1 (2023), p. 223.

[122] S. Cheng, Z. Li, R. Gao, B. Xing, Y. Gao, Y. Yang, S. Qin, L. Zhang, H. Ouyang, P. Du, et al. "A pan-cancer single-cell transcriptional atlas of tumor infiltrating myeloid cells". In: *Cell* 184.3 (2021), pp. 792–809.

[123] O. Franzén, L.-M. Gan, and J. L. M. Björkegren. "PanglaoDB: a web server for exploration of mouse and human single-cell RNA sequencing data". In: *Database* 2019 (Apr. 2019), baz046. ISSN: 1758-0463. DOI: 10.1093/database/baz046. eprint: https://academic.oup.com/database/article-pdf/doi/10.1093/database/baz046/28277084/baz046.pdf. URL: https://doi.org/10.1093/database/baz046.

[124] T. Liu, K. Li, Y. Wang, H. Li, and H. Zhao. "Evaluating the Utilities of Large Language Models in Single-cell Data Analysis". In: *bioRxiv* (2023), pp. 2023–09.

[125] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[126] J. Kim, Z. Xu, and P. A. Marignani. "Single-cell RNA sequencing for the identification of early-stage lung cancer biomarkers from circulating blood". In: *NPJ Genomic Medicine* 6.1 (2021), p. 87.

[127] A. A. Alizadeh, V. Aranda, A. Bardelli, C. Blanpain, C. Bock, C. Borowski, C. Caldas, A. Califano, M. Doherty, M. Elsner, et al. "Toward understanding and exploiting tumor heterogeneity". In: *Nature medicine* 21.8 (2015), pp. 846–853.

[128] M. Sorin, M. Rezanejad, E. Karimi, B. Fiset, L. Desharnais, L. J. Perus, S. Milette, M. W. Yu, S. M. Maritan, S. Doré, et al. "Single-cell spatial landscapes of the lung tumour immune microenvironment". In: *Nature* 614.7948 (2023), pp. 548–554.

[129] R. Peyser, S. MacDonnell, Y. Gao, L. Cheng, Y. Kim, T. Kaplan, Q. Ruan, Y. Wei, M. Ni, C. Adler, et al. "Defining the activated fibroblast population in lung fibrosis using single-cell sequencing". In: *American journal of respiratory cell and molecular biology* 61.1 (2019), pp. 74–85.

[130] A. M. Van der Leun, D. S. Thommen, and T. N. Schumacher. "CD8+ T cell states in human cancer: insights from single-cell analysis". In: *Nature Reviews Cancer* 20.4 (2020), pp. 218–232.

[131] Y. A. Reshef, L. Rumker, J. B. Kang, A. Nathan, I. Korsunsky, S. Asgari, M. B. Murray, D. B. Moody, and S. Raychaudhuri. "Co-varying neighborhood analysis identifies cell populations associated with phenotypes of interest from single-cell transcriptomics". In: *Nature biotechnology* 40.3 (2022), pp. 355–363.

[132] M. Sade-Feldman, K. Yizhak, S. L. Bjorgaard, J. P. Ray, C. G. de Boer, R. W. Jenkins, D. J. Lieb, J. H. Chen, D. T. Frederick, M. Barzily-Rokni, et al. "Defining T cell states associated with response to checkpoint immunotherapy in melanoma". In: *Cell* 175.4 (2018), pp. 998–1013.

[133] F. Zhang, A. H. Jonsson, A. Nathan, N. Millard, M. Curtis, Q. Xiao, M. Gutierrez-Arcelus, W. Apruzzese, G. F. Watts, D. Weisenfeld, et al. "Deconstruction of rheumatoid arthritis synovium defines inflammatory subtypes". In: *Nature* 623.7987 (2023), pp. 616–624.

[134] J. P. Engelmann, A. Palma, J. M. Tomczak, F. Theis, and F. P. Casale. "Mixed Models with Multiple Instance Learning". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2024, pp. 3664–3672.

[135] A. Litinetskaya, M. Shulman, S. Hediyeh-zadeh, A. A. Moinfar, F. Curion, A. Szalata, A. Omidi, M. Lotfollahi, and F. J. Theis. "Multimodal weakly supervised learning to identify disease-specific changes in single-cell atlases". In: *bioRxiv* (2024), pp. 2024–07.

[136] T. Liu, E. De Brouwer, T. Kuo, N. Diamant, A. Missarova, H. Wang, M. Hao, H. C. Bravo, G. Scalia, A. Regev, et al. "Learning multi-cellular representations of single-cell transcriptomics data enables characterization of patient-level disease states". In: *International Conference on Research in Computational Molecular Biology*. Springer. 2025, pp. 303–306.

[137] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. "Deep sets". In: *Advances in neural information processing systems* 30 (2017).

[138] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh. "Set transformer: A framework for attention-based permutation-invariant neural networks". In: *International conference on machine learning*. PMLR. 2019, pp. 3744–3753.

[139] M. Ilse, J. Tomczak, and M. Welling. "Attention-based deep multiple instance learning". In: *International conference on machine learning*. PMLR. 2018, pp. 2127–2136.

[140] J. Kim, J. Yoo, J. Lee, and S. Hong. "Setvae: Learning hierarchical composition for generative modeling of set-structured data". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15059–15068.

[141] H. Fan, H. Su, and L. J. Guibas. "A point set generation network for 3d object reconstruction from a single image". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 605–613.

[142] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[143] A. Q. Nichol and P. Dhariwal. "Improved denoising diffusion probabilistic models". In: *International conference on machine learning*. PMLR. 2021, pp. 8162–8171.

[144] X. Zheng, X. Huang, G. Mei, Y. Hou, Z. Lyu, B. Dai, W. Ouyang, and Y. Gong. "Point Cloud Pre-training with Diffusion Models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 22935–22945.

[145] C. V. Theodoris, L. Xiao, A. Chopra, M. D. Chaffin, Z. R. Al Sayed, M. C. Hill, H. Mantineo, E. M. Brydon, Z. Zeng, X. S. Liu, et al. "Transfer learning enables predictions in network biology". In: *Nature* 618.7965 (2023), pp. 616–624.

[146] A. Gayoso, Z. Steier, R. Lopez, J. Regier, K. L. Nazor, A. Streets, and N. Yosef. "Joint probabilistic modeling of single-cell multi-omic data with totalVI". In: *Nature methods* 18.3 (2021), pp. 272–282.

[147] C. De Donno, S. Hediyeh-Zadeh, A. A. Moinfar, M. Wagenstetter, L. Zappia, M. Lotfollahi, and F. J. Theis. "Population-level integration of single-cell datasets enables multi-scale analysis across samples". In: *Nature Methods* 20.11 (2023), pp. 1683–1692.

[148] Y. Mao, Y.-Y. Lin, N. K. Wong, S. Volik, F. Sar, C. Collins, and M. Ester. "Phenotype prediction from single-cell RNA-seq data using attention-based neural networks". In: *Bioinformatics* 40.2 (2024), btae067.

[149] B. He, M. Thomson, M. Subramaniam, R. Perez, C. J. Ye, and J. Zou. "Cloudpred: Predicting patient phenotypes from single-cell rna-seq". In: *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2022*. World Scientific. 2021, pp. 337–348.

[150] B. v. Querfurth, J. Lohmoeller, J. Pennekamp, T. Bleckwehl, R. Kramann, K. Wehrle, and S. Hayat. "mcBERT: Patient-Level Single-cell Transcriptomics Data Representation". In: *bioRxiv* (2024), pp. 2024–11.

[151] G. Corso, H. Stärk, B. Jing, R. Barzilay, and T. Jaakkola. "Diffdock: Diffusion steps, twists, and turns for molecular docking". In: *arXiv preprint arXiv:2210.01776* (2022).

[152] E. Luo, M. Hao, L. Wei, and X. Zhang. "scDiffusion: conditional generation of high-quality single-cell data using diffusion model". In: *Bioinformatics* 40.9 (2024), btae518.

[153] J. Liu, Y. Pan, Z. Ruan, and J. Guo. "SCDD: a novel single-cell RNA-seq imputation method with diffusion and denoising". In: *Briefings in Bioinformatics* 23.5 (2022), bbac398.

[154] S. Dong, Z. Cui, D. Liu, and J. Lei. "scRDiT: Generating single-cell RNA-seq data by diffusion transformers and accelerating sampling". In: *arXiv preprint arXiv:2404.06153* (2024).

[155] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. "Film: Visual reasoning with a general conditioning layer". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1. 2018.

[156] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[157] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[158] T. Li, Y. Tian, H. Li, M. Deng, and K. He. "Autoregressive image generation without vector quantization". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 56424–56445.

[159] e. a. CZI Single-Cell Biology. "CZ CELLxGENE Discover: A single-cell data platform for scalable exploration, analysis and modeling of aggregated data". In: *bioRxiv* (Oct. 2023). DOI: 10.1101/2023.10.30.563174. URL: https://doi.org/10.1101/2023.10.30.563174.

[160] L. McInnes, J. Healy, and J. Melville. "Umap: Uniform manifold approximation and projection for dimension reduction". In: *arXiv preprint arXiv:1802.03426* (2018).

[161] L. Sikkema, C. Ramírez-Suástegui, D. C. Strobl, T. E. Gillett, L. Zappia, E. Madissoon, N. S. Markov, L.-E. Zaragosi, Y. Ji, M. Ansari, et al. "An integrated cell atlas of the lung in health and disease". In: *Nature Medicine* (2023), pp. 1–15.

[162] R. K. Perez, M. G. Gordon, M. Subramaniam, M. C. Kim, G. C. Hartoularos, S. Targ, Y. Sun, A. Ogorodnikov, R. Bueno, A. Lu, et al. "Single-cell RNA-seq reveals cell type–specific molecular and genetic associations to lupus". In: *Science* 376.6589 (2022), eabf1970.

[163] E. Stephenson, G. Reynolds, R. A. Botting, F. J. Calero-Nieto, M. D. Morgan, Z. K. Tuong, K. Bach, W. Sungnak, K. B. Worlock, M. Yoshida, et al. "Single-cell multi-omics analysis of the immune response in COVID-19". In: *Nature medicine* 27.5 (2021), pp. 904–916.

[164] L. Atanackovic, X. Zhang, B. Amos, M. Blanchette, L. J. Lee, Y. Bengio, A. Tong, and K. Neklyudov. "Meta flow matching: Integrating vector fields on the wasserstein manifold". In: *arXiv preprint arXiv:2408.14608* (2024).

[165] D. Hendrycks and K. Gimpel. "Gaussian error linear units (gelus)". In: *arXiv preprint arXiv:1606.08415* (2016).

[166] Z. Wu, A. E. Trevino, E. Wu, K. Swanson, H. J. Kim, H. B. D'Angio, R. Preska, G. W. Charville, P. D. Dalerba, A. M. Egloff, et al. "SPACE-GM: geometric deep learning of disease-associated microenvironments from multiplex spatial protein profiles". In: *bioRxiv* (2022), pp. 2022–05.

[167] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu. "Do transformers really perform badly for graph representation?" In: *Advances in neural information processing systems* 34 (2021), pp. 28877–28888.

[168] E. Kernfeld, Y. Yang, J. S. Weinstock, A. Battle, and P. Cahan. "A systematic comparison of computational methods for expression forecasting". In: *BioRxiv* (2023), pp. 2023–07.

[169] C. Ahlmann-Eltze, W. Huber, and S. Anders. "Deep learning-based predictions of gene perturbation effects do not yet outperform simple linear methods". In: *BioRxiv* (2024), pp. 2024–09.