Probabilistic Algorithms for Integrated Analysis of Single-Cell Multi-Omic Data

by

Pinar Demetci

B.Sc., Olin College of Engineering, MA, United States, 2017

A dissertation submitted in partial fulfillment of the
requirements for the Degree of Doctor of Philosophy
in the Department of Computer Science and Center for Computational Molecular Biology
at Brown University

Providence, Rhode Island

May 2023

This dissertation by Pinar Demetci is accepted in its present form by the Department of Computer Science and Center for Computational Molecular Biology as satisfying the dissertation requirement for the degree of Doctor of Philosophy.

Date _____ _____

Dr. Ritambhara Singh, Primary Advisor
(Computer Science)

Date _____ _____

Dr. Sorin Istrail, Secondary Advisor
(Computer Science)

Recommended to the Graduate Council

Date _____ _____

Dr. Lorin Crawford, Reader
(Biostatistics, Microsoft Research)

Date _____ _____

Dr. Erica Larschan, Reader
(Molecular Biology, Cell Biology, and Biochemistry)

Approved by the Graduate Council

Date _____ _____

Dr. Thomas A. Lewis
Interim Dean of the Graduate School

Vita

PINAR DEMETCI

Portfolio: pinardemetci.github.io ◇ **Google Scholar:** tinyurl.com/DemetciScholar

RESEARCH INTERESTS

Methods: representation learning, manifold learning, optimal transport, geometric deep learning, dynamic neural networks, interpretability in ML/DL, Bayesian inference, causal inference.

Application domains: regulatory genomics, precision medicine, single-cell data science, multi-modal data integration, drug discovery, cellular reprogramming, cancer, neurodenegeration.

EDUCATION

Brown University Providence, RI
Ph.D. Computer Science, Computational Biology — GPA: 3.90/4.00 **2023**
Advisors: Ritambhara Singh. Ph.D. (primary), Sorin Istrail, Ph.D. (secondary)
M.Sc. Computer Science — *(Concurrent degree)* GPA: 4.0/4.0
Advisor: Sorin Istrail, Ph.D.

Olin College of Engineering Needham, MA
B.Sc. Engineering (concentration: Bioengineering) — GPA: 3.67/4.00 **2017**

RESEARCH EXPERIENCE

Brown University Sept 2018 - Present
Graduate Research Assistant Providence, RI

- Developing probabilistic algorithms for integrated analysis of single-cell multi-modal data, with applications in regulatory genomics. [Publications #4,#6,#7,#8].
- Previously worked on (1) a multi-scale Bayesian variable selection method in neural networks with applications in genome-wide association studies [Publication #5], and (2) combinatorial algorithms for gene expression prediction from haplotype sequences [Publication #3].

Microsoft Research June 2022 - August 2022
Research Intern Redmond, WA & Cambridge, MA (remote)

- Developed a new computational method based on optimal transport with metric learning through deep learning classifiers to predict mechanism of action of chemical and genetic perturbations from single-cell transcriptomic datasets.

Microsoft Research June 2020 - September 2020
Research Intern Redmond, WA (remote)

- Implemented a Bayesian inference and machine learning pipeline on Microsoft Azure for disease risk prediction and clinical and genomic marker identification.

Massachusetts Institute of Technology June 2017 - August 2018
Research Support Associate (Full Time) Cambridge, MA

· Served as the lab manager in Gene-Wei Li lab. Investigated bacterial regulatory network rewiring [Publication #2]. Contributed to various quantitative biology experiments and data analysis pipelines in the lab.

Olin College of Engineering Sep 2015 - May 2017
Undergraduate Research Assistant Needham, MA

· Worked with Drs. Jean Huang and John Geddes on Lotka-Volterra-based dynamic modeling for bacterial communities undergoing environmental perturbations.

· Designed and implemented an assistive software prototype for blind navigation in Olin College Crowdsourcing and Machine Learning (OCCaM) Lab with Dr. Paul Ruvolo.

Daktari Diagnostics Jan 2016 - Dec 2016
Student Engineer Cambridge, MA

· Implemeted an image analysis software to automate diagnosis for a novel rapid sickle-cell diagnostic device. Worked as a technical lead in a team of five engineering students.

AWARDS AND HONORS

2023	Eric-Wendy Schmidt Postdoctoral Fellowship (Broad Institute of MIT and Harvard)
2023	Dana Farber Data Science Fellowship *(turned down)*
2022	Rising Stars in Electrical Engineering and Computer Science (by UT Austin)
2022	RECOMB Travel Fellowship
2020	Microsoft Research Ph.D. Fellowship Nominee (by Brown CCMB)
2020	ICML WCB Fellowship
2020	ICML WCB Best Poster Award
2016	Meritorious Winner (Top 10%): COMAP MCM/ICM Contest in Mathematical Modeling
2015-2017	Olin Alumni Merit Scholarship (towards living expenses)
2013-2017	Sunlin Chou International Scholarship (50% tuition)
2013-2017	Olin Merit Scholarship (50% tuition)
2013	Honorable Mention, the 21^{st} Intl. Competition of First Step to Nobel Prize in Physics by the Polish Academy of Sciences
2013	First Place in Physics, the 22^{nd} MEF International Research Projects Contest
2008-2013	Turkish Education Foundation (TEV) Scholarship for Gifted Students

PEER-REVIEWED PUBLICATIONS

(*) Denotes co-first authors, Undergraduate mentees underlined

9. **P Demetci**, Q. H. Tran, I Redko, R Singh. (2022). Jointly aligning cells and features of single-cell multi-omics datasets with co-optimal transport. *NeurIPS Learning Meaningful Representations of Life (LMRL)* and *Machine Learning in Computational Biology (MLCB)*

8. QH Tran, H Janati, N Courty, R Flamary, I Redko, **P Demetci**, R Singh. (2022). Unbalanced CO-Optimal Transport. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2023).*

7. **P Demetci**, R Santorella, B Sandstede, R Singh. (2022). Unsupervised integration of single-cell multi-omics datasets with disparities in cell type representation. *International*

Conference on Research in Computational Biology (RECOMB 2022). Springer Lecture Notes in Computer Science. 306:3-19.

- Also appeared as an extended paper:
  P Demetci, R Santorella, M Chakravarthy, B Sandstede, W Stafford Noble, R Singh. (2022). SCOTv2: Single-cell multi-omic alignment with disproportionate cell-type representation. *Journal of Computational Biology.* RECOMB 2022 issue.

6. **P Demetci***, R Santorella*, B Sandstede, W Stafford Noble, R Singh. (2021). Gromov-Wasserstein optimal transport to align single-cell multi-omics data *International Conference on Research in Computational Molecular Biology (RECOMB 2021).*
   - Also published as an extended paper:
     P Demetci, R Santorella, B Sandstede, W Stafford Noble, and R Singh. (2021). SCOT: Single-Cell multi-omic integration with Optimal Transport *Journal of Computational Biology.* 29(1):3-18. RECOMB 2021 issue
   - Also appeared as an abstract at: *the 37th International Comference on Machine Learning (ICML) Workshop on Computational Biology (2020).*
5. **P Demetci***, W Cheng*, G Darnell, X Zhou, S Ramachandran, L Crawford. (2021) Multi-scale genomic inference using biologically annotated neural networks. *PLOS Genetics.* 17(8): e1009754.
4. R Singh, **P Demetci**, G Bonora, V Ramani, C Lee, H Fang, Z Duan, X Deng, J Shendure, C Disteche, W Stafford Noble. (2020) Unsupervised manifold alignment for single-cell multi-omics data *ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (ACM BCB)* .
3. B Alpay*, **P Demetci***, S Istrail, D Aguiar. (2020) Combinatorial and statistical prediction of gene expression from haplotype sequence. *Bioinformatics.* 36 (Supp-1): i194-i202.
2. D Parker*, **P Demetci***, G W Li. (2019) Rapid accumulation of motility-activating mutations in resting liquid culture of *Escherichia coli*. *Journal of Bacteriology.* 201(19):e00259.
1. **P Demetci**, C Nichols, Y V Zastavker, J D Stolk, A Dillon, M Gross. (2016) Externalization and internalization in the classroom: How do they emerge and why is it important? *IEEE Frontiers in Education Conference.*

SELECTED CONFERENCES AND INVITED TALKS

(Presentations given outside my home institution are listed only. "Acc.R.": Acceptance Rate.)

Title: "Unbalanced CO-Optimal Transport"

2023 The AAAI Conference on Artificial Intelligence (AAAI): Poster

Title: "Joint alignment of cells and genomic features of single-cell multi-omic datasets with co-optimal transport"

2022 Machine Learning in Comp. Bio. (MLCB): Talk & Poster *(Acc.R.:* 17%*)*

2022 CSHL Biological Data Science Meeting: Poster

2022 NeurIPS Learning Meaningful Representations of Life: Talk & Poster

Title: "Unsupervised integration of single-cell multi-omics datasets with disparities in cell-type representation"

2022 RECOMB Proceedings: Talk *(Acc.R.:* 20%*)*
RECOMB: International Conference on Research in Computational Molecular Biology

2021 Machine Learning in Comp. Bio. (MLCB): Contributed Talk *(Acc.R.:* 25%*)*

Title: "Enabling integrated analysis of single-cell multi-omics data with optimal transport"

2022 SigmaXi iFoRE Conference: Invited Talk
("The Convergence of Data, Geometry, and Biology: Insights from the 'shape' of biological data" session)

2021 NeurIPS Workshop on Optimal Transport in ML: Invited Keynote Talk

2021 University of Connecticut Bioinformatics Seminar: Invited Seminar Talk

Title: "Biologically Annotated Neural Networks for Multi-Scale Genomic Discovery in GWAS"

2022 Stanford University (Kundaje Lab): Invited Seminar Talk

Title: "Gromov-Wasserstein Optimal Transport to Align Single-cell Multi-omics Data"

2021 RECOMB Proceedings: Contributed Talk *(Acc.Rate: 19%)*

2021 Machine Learning in Comp. Bio.(MLCB): Contributed Talk *(Acc.R.: 15%)*

2020 Workshop on Optimal Control, Optimal Transport & Data Science at the U. of Minnesota – Institute for Mathematics and Its Applications: Poster

2020 ICML Workshop on Comp. Bio.: Spotlight Talk & Poster *(Acc.R.:* 21%*)*

2020 Intelligent Systems in Molecular Bio. (ISMB - MLCB track): Contributed Talk & Poster *(Acc.R.:* 25%*)*

TECHNICAL SKILLS

Languages	Python, R, Java, C++, MATLAB, SQL *(ordered by competency level)*
Environments	Linux, UNIX, slurm, High Performance Computing, Google Cloud, Azure
Frameworks & Tools	***Machine Learning:*** PyTorch, PyTorch-Geometric (PyG), Tensorflow, DGL, Pyro, numPyro, tf-probability, CUDA, Python OT (POT), Sklearn, SkImage ***Data Processing:*** Numpy, Pandas, SciPy, HDF5, AnnData, openCV, Pillow ***Data Visualization:*** ggplot, matplotlib, seaborn, Bokeh, D3 ***Bioinformatics:*** ScanPy, Seurat, Signac, Bioconductor, VCFtools, BEDtools, SAMtools, PLINK, Cromwell, GATK, Picard PyMOL, Cell Profiler, Cytoscape, QIIME, VAMPS ***Other:*** Cython, Armadillo, Docker

PROFESSIONAL COMMUNITY SERVICE & MEMBERSHIPS

Reviewing:

2023	Reviewer for NAR Genomics & Bioinformatics, and Bulletin of Mathematical Bio.
2023	Program Committee member for ISMB/ECCB Abstracts (MLCSB COSI)
2022	Program Committee member for the MLCB conference
2020 - 2022	Reviewer for MLCB 2020 – 2022 conferences
2020 - 2023	Sub-reviewer for ICML, NeurIPS, RECOMB conferences

Mentorship:

2022 - Present	Alumnus Mentor for the "Olin College Banter Program"
2020 - 2021	Mentor for the "Application Feedback Program for Under-represented Applicants" in the Computer Science Department at Brown University
2020 - Present	Peer Mentor for International Graduate Students at Brown University

Admissions:

2020 - 2021 Brown U. Computational Biology Ph.D. Program admissions committee member

Memberships:

2020 - Present	Student Member at Society for Industrial and Applied Mathematics (SIAM) AnitaB.org, and New England Statistical Society (NESS).
2018 - Present	Student Member at International Society for Computational Biology (ISCB)
2018 - 2023	Member at Graduate Women in Science and Engineering (GWiSE) at Brown U.

TEACHING EXPERIENCE

Instructor at Brown University

- **(Workshop) Intro. to High Performance and Parallel Computing** (Fall 2022)
  Organized and led a hands-on workshop series for training first year graduate students and T32 trainees on high performance and parallel computing.

Graduate Teaching Assistant at Brown University

- **CSCI 2820 Advanced Algorithms in Computational Biology & Medical Bioinformatics** (Spring 2019 & 2021)
  Lectured twice, held recitations and office hours, wrote and graded assignments
  Instructor: Sorin Istrail, Ph.D.

Undergraduate Teaching Assistant at Olin College of Engineering

- **SCI1240: Designing Better Drugs, with Laboratory** (Fall 2015)
  Assisted in the classroom and laboratory, graded assignments
  Instructor: Joanne Pratt, Ph.D.

RESEARCH MENTORSHIP

Project: *"Interpretable graph neural networks for cell-type prediction using single-cell multi-omic data"*

Mentees:

- Hossam Zaki (Class of 2022, Brown University, Senior Honors Thesis) → now MD-PhD student at Brown
- Momoka Kobayashi (Brown University, Class of 2023, Biomedical Engineering and Computer Science)

Project: *"Optimal transport for identify cell-type similarities across mammalian and reptilian brain cortices"*

Mentees:

- Samantha Hong (Brown U. Class of 2023, Computational Biology)
- Manav Chakravarthy (Brown University, Class of 2024, Computer Science and Economics)

Acknowledgements

I am lucky to have many people to thank for the support and guidance they provided me during (and on my path to) my doctoral studies.

First of all, I feel genuinely fortunate to be advised by Dr. Ritambhara Singh. I look up to Ritambhara, not only as a great role-model of a scientist with a strong work ethic, but also as an effective mentor and a delightful person to work with and learn from. In the four years that I have worked with her, I consistently felt that I had my advisor's support, no matter what the situation was. In good times, she celebrated our accomplishments with excitement and nominated our work for workshops that were helpful for my professional development. In challenging times, she was encouraging, kept me optimistic, and offered her time, guidance, and critical feedback. I felt that she always valued my opinion, even when it differed from hers. Her solution-oriented communication style and empathetic leadership made me feel comfortable discussing any challenges that arose. I admire her optimistic personality, the influential research program she has launched, and the the inclusive lab environment she has created. My experience with her contributed to my enjoyment of research, and I think that is the most significant impact an advisor can make. Many thanks to you, Ritambhara, for being an excellent mentor for me.

I have been very lucky to have a second highly supportive advisor, Dr. Sorin Istrail. It has been a privilege to have an advisor who has been so influential in establishing our field, starting from his leadership in the development of the genome assembly algorithms for the Human Genome Project, to his role in launching the RECOMB conference, where two of the algorithms presented in this thesis were published. I find Sorin's life story and

career trajectory to be highly inspiring as an immigrant from a somewhat proximal part of the world. From a personal standpoint, I always looked forward to my meetings with Sorin, because all his interactions with his students are filled with sincerity and warmth. I would like to thank Sorin for the enthusiastic support he gave me all throughout my PhD, and also for always making time to meet with me even when he was on a sabbatical. Above all, I want to thank both of my advisors for encouraging me to believe in my place in science. From an academic standpoint, Sorin's advising has been complimentary to Ritambhara's in certain aspects: Sorin and I got to work on combinatorial problems, and merging combinatorics with statistics, which was quite different than Ritambhara's research focus. Sorin encouraged me to dig deep into the mathematical foundations and challenge the assumptions made in existing computational approaches. This led to several meetings in front of the whiteboard, driving theorems, and it also encouraged me to pick up textbooks on subjects I was taking for granted. Sorin also introduced me to Derek Aguiar, who has been another (unofficial) advisor to me. Derek was Sorin's previous Ph.D. student, and is now a professor at UConn. My first Ph.D. publication (not included in this thesis due to its differing focus) was with Derek and Sorin. Since then, Derek has continued to include me in his lab meetings and project discussions, which involve Bayesian machine learning and its applications in computational genomics. So, I also want to thank Derek for not only collaborating with me, but also for treating me just like one of his own graduate students, who are very lucky to be in his lab.

Next, I would like to thank my thesis committee members, Dr. Lorin Crawford and Dr. Erica Larschan. Lorin's role in my Ph.D. goes beyond being a committee member; he was also an early advisor in my Ph.D. The research project we worked on together (not included in this thesis) was my introduction to computational methods development. I learned most of what I know about statistical genetics and Bayesian inference through this project from Lorin. I am grateful that he continued to be a mentor for me by being a part of my thesis committee even after we completed our work together. I am also grateful to him for always being available to give his perspective on my career-related questions. I

admire his dynamism and creativity in research. As a program chair, Erica's impact on my Ph.D. also goes beyond her role in my thesis committee. I think we are truly lucky to have a chair like her, who is so genuinely invested in creating a welcoming environment for all students. Erica considers students' best interest in her decision making and her support helped me to navigate some challenging times in my graduate experience. She is easily the most enthusiastic person I know when it comes to discussing students' work: I have always left our discussions feeling more motivated for research. I thank Erica for her valuable contributions to this thesis through the feedback she provided with her expertise in biology, as well as for the unwavering encouragement she gave me throughout my Ph.D. studies.

The work in this thesis would not have been possible without my collaborators. I especially would like to thank Dr. Rebecca Santorella and Dr. Ievgen Redko, who have been the most delightful collaborators to work with. Rebecca worked on the "SCOT" project with me as a Ph.D. student in applied mathematics, and I could easily say that was one of the most enjoyable times in my Ph.D. Ievgen is another scientist whose positive personality I admire in addition to looking up to his deep expertise in computational optimal transport. He has been another effective advisor for me on the "SCOOTR" project. I also thank Rebecca's Ph.D. advisor, Dr. Bjorn Sandstede, and Ritambhara's postdoctoral advisor, Dr. William Stafford Noble, for their advising on the "SCOT" project, Quang Huy Tran for collaborating on "SCOOTR" and unbalanced co-optimal transport, and lastly, Dr. Remi Flamary and Dr. Nicolas Courty for useful discussions of optimal transport applications.

The community members and staff at CCMB, as well as my friends both from and outside of Brown have been an integral part of my life these past five years. I thank Nathaniel Gill and Maria Cardone for going above and beyond to increase our quality of life at CCMB and also for making all administrative work go so smoothly. I want to thank Dr. Emilia Huerta-Sanchez, who has been another supportive faculty member for me at CCMB. Due to the difference in our research focus, we never got to work together; nontheless, she has given me such valuable career advice and let me know that her door was always open if I had anything to discuss. Among my friends, I especially thank Melisa,

Ken, Jiaying, James, Dana, Rachel, Kexin, Jeremy, Yusuke, (An)Dressa, Irene, Chib, Ria, Marjan, Aaron, Amina, Murtaza, Mayra, Hyeyeon, Vivek, Leah, Julian, Cecile, Cole, Qing, Jiaqi, Atishay, Michal, Sam, Abi, Halley, Juanita, Wei, George, Wasiwasi, Ashley, Ananya, Tuan, and Alex for their support and friendship.

I would not have had the opportunity to pursue this doctorate had I not received the support of several people during my undergraduate education. I moved to the United States from Turkey to attend my undergraduate institution thanks to the generous financial support Dr. Sunlin Chou provided. Dr. Chou had also moved to the U.S. as an international student from Singapore to attend MIT. As a trustee at Olin, when he wanted to support international students in similar positions, the previous head of admissions, Charles Nolan, highlighted my application. I want to thank them both for allowing me to embark on this life-changing experience that otherwise would not have been possible. Dr. Chou passed away in December 2017, but I will forever feel grateful for his support. I discovered the field of computational biology at Olin, through my research experience with Dr. Jean J Huang and Dr. John Geddess, working on mathematical modeling of bacterial communities under various perturbations. Then, I took Dr. Brian Tjaden's computational biology course at Wellesley and developed a passion for computer science through the classes I took and the research projects I worked on with Dr. Paul Ruvolo at Olin. I want to thank them all for helping me find my calling in computational biology. I especially want to thank Jean for giving me encouragement when I decided to apply to graduate programs. I also thank Dr. Gene-Wei Li at MIT, who took a chance on me and hired me as a research associate in his quantitative biology lab after college. I am grateful that he still keeps in touch and checks up on me. These professors have been some of the most supportive mentors I know, and I feel very lucky to have worked with them.

Finally, I want to express my gratitude to my family, specifically my father Mehmet, mother Nuran, and older siblings Hasan Cem and Esen, for their constant and unconditional support. Despite being halfway across the world, they never let me feel alone. I want to thank my mom for the sacrifice she has made when she let me move to the U.S. even

though she was tempted to protest it, my father Mehmet for enthusiastically supporting all my endeavours and having such solid faith in me, my brother Hasan Cem for always being there for me as my rock, and also for traveling with my mom for my graduation, my sister Esen for always being available to chat and give me emotional support. Additionally, I am grateful to my extended family members and family friends, such as my aunts, uncles and Serpil, for their steadfast support. I dedicate this thesis to my father, who has been the greatest role model I have had as the most hardworking yet humble person I know.

-Pinar Demetci.

Abstract of "Probabilistic Algorithms for Integrated Analysis of Single-Cell Multi-Omic Data" by Pinar Demetci, Ph.D., Brown University, May 2023.

Advances in sequencing technologies in the last decade have enabled us to profile various genomic features at the single-cell resolution, such as gene expression and chemical modifications on the DNA. Studying how these features co-vary across cells can reveal how they interact to regulate cellular processes across cell types and states. However, with some exceptions, it is not possible to simultaneously take multiple types of genomic measurements on the same cells due to the destructive nature of sequencing technologies. Multi-modal studies of single-cell genomes thus require computational methods that integrate data from different sequencing experiments.

This dissertation presents three probabilistic algorithms designed to address certain real-world challenges that existing algorithms fail to address when integrating various single-cell measurements. The first one, *Single-Cell alignment with Optimal Transport* (SCOT), is an unsupervised algorithm that compares dataset geometries to yield probabilistic cell alignments between two datasets. When there is validation data available for hyperparameter tuning, SCOT gives results on par with the state-of-the-art alignment algorithms. Unlike these algorithms, however, SCOT heuristically self-tunes its hyperparameters and still yields high-quality alignments when users do not have sufficient validation data. This is a realistic scenario as different features are profiled in different cells in single-cell experiments. The second algorithm, *SCOT version 2* (SCOTv2), extends SCOT to align more than two datasets at a time. It also handles datasets with disproportionate cell-type representation, which we show is a common phenomenon in real-world experiments that most alignment algorithms fail to account for. The third algorithm, *Single-cell fused Gromov CO-Optimal TRansport* (SCOOTR), uses a novel optimal transport formulation to jointly align both cells and features through an alternating optimization scheme. This joint formulation not only improves cell alignments but also generates hypotheses about the relationships between

genomic features. SCOOTR additionally allows for users to provide weak supervision on either the feature or the cell alignments in order to improve the quality of both.

Contents

List of Tables **xx**

List of Figures **xxii**

1 Introduction **1**

1.1 Background on Single-cell Multi-omics 1

1.1.1 Genome regulation and sequencing 1

1.1.2 Single-cell sequencing and multi-omics 2

1.2 Thesis Overview and Summary of Contributions 7

2 Background on Optimal Transport **10**

2.1 Definitions and Notations 10

2.2 Overview: The Monge Problem and the Kantorovich Relaxation . . . 12

2.3 Efficient Computational Solvers and Associated Regularizers 15

3 SCOT: Unsupervised Single-cell alignment with (Gromov- Wasserstein) Optimal Transport **18**

3.1 Introduction 18

3.1.1 Existing Approaches (as of the acceptance for publication in 2020) 20

3.1.2 Our contributions 21

3.2 Methods 23
3.2.1 Optimal Transport 24
3.2.2 Gromov-Wasserstein Optimal Transport 25
3.2.3 Single-Cell alignment using Optimal Transport (SCOT) 27
3.2.4 Alternative Unsupervised Alignment Procedure 28
3.3 Experimental Setup 31
3.3.1 Simulated datasets 31
3.3.2 Single-cell multi-omics datasets 31
3.3.3 Evaluation metrics 32
3.3.4 Hyperparameter tuning 34
3.4 Results 35
3.4.1 SCOT successfully aligns the simulated datasets 35
3.4.2 SCOT gives state-of-the-art performance for single-cell multi-omics alignment 36
3.4.3 SCOT's alternative unsupervised hyperparameter tuning procedure achieves quality alignments 39
3.4.4 SCOT's computation speed scales well with the sample size . . 40
3.4.5 Investigating algorithmic choices and hyperparameters of SCOT 41
3.5 Discussion 43

4 SCOTv2: Unbalanced Multi-domain Single-cell Alignment 45

4.1 Introduction 45
4.2 Our contributions 46
4.3 Method 48
4.3.1 Unbalanced Optimal Transport of SCOTv2 49
4.3.2 Extending SCOTv2 for Multi-Domain Alignment 50
4.3.3 Embedding with the Coupling Matrix 51
4.3.4 Embedding Method Details 52

4.3.5 Heuristic process for self-tuning hyperparameters 55
4.4 Experimental Setup 56
4.4.1 Datasets 56
4.4.2 Evaluation metrics and baseline methods 60
4.5 Hyperparameter Tuning Procedure Details 61
4.6 Results 63
4.6.1 SCOTv2 gives high-quality alignments consistently across all single-datasets 63
4.6.2 Hyperparameter self-tuning aligns well without depending on orthogonal correspondence information 67
4.6.3 SCOTv2 scales well with increasing number of samples 68
4.7 Discussion 68

5 Jointly aligning samples and features of datasets 72
5.1 Introduction 72
5.1.1 Related Works 73
5.1.2 Our contributions 73
5.2 Method 74
5.2.1 Single-cell fused Gromov Co-Optimal Transport (SCOOTR) . 76
5.2.2 Providing supervision to SCOOTR 77
5.3 Experimental Setup 79
5.3.1 Datasets 79
5.3.2 Evaluation Criteria 82
5.3.3 Baselines 83
5.4 Results 84
5.4.1 SCOOTR improves upon cell alignment performance of SCOT and SCOTv2 84

5.4.2 SCOOTR generates biologically meaningful hypotheses on feature correspondences 86
5.4.3 Supervision on one level (e.g. cell- or feature-level alignments) improves alignment quality of both 87
5.5 Discussion 91

6 Conclusion **94**

7 Appendix **98**
7.1 Proofs for SCOOTR 98
7.2 Python implementation of the proposed algorithms 102
7.3 Other relevant work published during Ph.D. program 128

⋆ **Disclaimer:** Parts of this dissertation appeared in proceedings of conferences or in journals. In particular, Chapter 3 is an extended version of [28] and [26], Chapter 4 is an extended version of [27] and [29], and finally, Chapter 5 is an extended version of [31]and [30].

List of Tables

3.1 Alignment performance by average FOSCTTM measure when the first domain is projected onto the second domain. For real-world datasets, we picked gene expression domain in scGEM and chromatin accessibility domain in SNAREseq to be projected. 36

3.2 Alignment performance by label transfer accuracy ($k = 5$) when the first domain (epigenomic domains in real-world datasets) is projected onto the second domain (gene expression domain in real-world datasets). 36

3.3 Best mean FOSCTTM for each direction of the barycentric projection for all datasets. The method is robust to the direction of the projection. 37

3.4 Alignment performance by mean FOSCTTM scores in fully unsupervised setting. The hyperparameters for SCOT are chosen by lowest Gromov-Wasserstein distance and the defaul hyperparameters are used for MMD-MA, and UnionCom. Best values are bolded. 40

3.5 Alignment performance by label transfer accuracy ($k = 5$) in the fully unsupervised setting when the first domain is used for training. The hyperparameters for SCOT are chosen by lowest Gromov-Wasserstein distance and the defaul hyperparameters are used for MMD-MA, and UnionCom. Best values are bolded. 40

4.1 Number of cells in (and percentages of) each cell-type across different modalities in the scNMT-seq co-assayed dataset after quality control procedures and the non-coassay datasets. 60

4.2 Alignment performance benchmarking in the fully unsupervised setting. We run SCOTv2 and SCOT using their heuristics to approximately self-tune hyperparameters. We use default parameters for other methods due to a lack of similar procedures for unsupervised self-tuning. 65

5.1 Quality of cell alignments yielded by SCOT, SCOOTR,and bindSC in the "balanced case" (no disproportionate cell-type representation across datasets), as quantified by the average FOSCTTM metric (lower values are better). 85

5.2 Quality of cell alignments yielded by SCOT, SCOOTR,and bindSC in the "unbalanced case" (disproportionate cell-type representation across datasets), as quantified by the label transfer accuracy metric (higher values are better). 85

5.3 Feature alignment performance on SNARE-seq and cross-species RNA-seq dataset with increasing supervision on cell-type alignments. . . . 90

5.4 Cell-type alignment performance on cross-species RNA-seq dataset with increasing supervision on paralogous gene alignments. 91

List of Figures

1.1 Overview of the eukaryotic genome architecture and some gene regulatory events. The figure is prepared on `Biorender.com` 3

1.2 A chart of genomic and cellular feature combinations for which at least one single-cell co-assaying technology is available (shown in green). For each combination, we give a *non-exhaustive* list of example protocols [16–20, 24, 32–34, 40, 41, 44, 45, 54, 58, 63, 64, 71, 73, 74, 76, 81, 85, 95, 100, 102, 103, 110]. This figure is adopted from [57] and updated with information from [91] to include new assaying protocols available in 2023. 6

2.1 Schematic of optimal transport maps in Monge's and Kantorovich's formulations. Optimal transport relates probabily distributions, either defined as (A) discrete measures or (B) continuous densities. (C) The Monge formulation seeks to find deterministic pushforward maps that will transport probability distributions onto each other. (D) Kantorovich's formulation relaxes the transport problem and allows for probabilistic transport maps. This figure is adapted from Peyre and Cuturi [70]. 12

2.2 **Visualizing the effect of entropic regularization on the optimal coupling.** As the entropic regularization coefficient increases, the coupling probabilities are more split, yielding a less sparse solution. In this example, we align two datasets with 10 MNIST handwritten digit samples [53] in each, as a toy dataset. 15

3.1 **Visualization of the single-cell multi-omic integration problem.** When experimentally co-profiling different aspects of the genome on single-cells is not possible, scientist apply different single-cell sequencing methods on separate aliquots of a cell population. This procedure yields disparate datasets as plotted, with limited to no prior information on 1-1 correspondences either between cells or features. . 20

3.2 **Schematic of SCOT alignment of single-cell multi-omics data.** A population of cells is aliquoted for different single-cell sequencing assays. SCOT constructs k-NN graphs based on sample-wise correlations and finds a probabilistic coupling between the samples of each domain that minimizes the distance between the two intra-domain graph distance matrices. Barycentric projection projects one domain onto another based on this coupling matrix. 23

3.3 Alignment results for simulated datasets.We present the alignment result on four simulations (left to right) - a bifurcation, a Swiss roll, a circular frustum, and synthetic RNA-seq data generated from Splatter [106] **A. Visualization of the dataset before alignment.** Each dataset has two domains to be aligned. **B. Visualization of datasets after alignment by SCOT.** The upper row plots samples colored by domain they come from, while the bottom row shows samples colored by their group (or cell-type) identity. **C. Performance benchmarking**. We plot sorted FOSCTTM measures for alignments performed by SCOT, MMD-MA, and UnionCom for benchmarking. Mean FOSCTTM measures for each alignment and dataset are included in figure legends. Best performing results are bolded. 38

3.4 Aligning real world single-cell sequencing dataset. **A.** We first visualize the original datasets before alignment. Each dataset has two domains with different sequencing modalities. **Left:** our alignment colored based by domain (plotted in 2D using PCA). **B.** We visualize the aligned datasets after running SCOT. For each dataset, we plot alignments both by coloring data points by domain and by cell-type identity. **C.** We benchmark SCOT against MMD-MA and UnionCom algorithms by comparing FOSCTTM values we get. Graphs here plot sorted FOSCTTM measures and the legend contains average FOSCTTM measures for each alignment. 39

3.5 A. Runtime comparisons with growing sample size. Dotted lines are polynomial trend lines. B. Relationship between Gromov-Wasserstein distance between the aligned datasets and alignment quality. Lower Gromov-Wasserstein values tend to correspond to better alignments (lower FOSCTTM measures). 41

3.6 Hyperparameter tuning results for scGEM (left) and SNARE-seq (right) datasets. We sweeped a range of values for the two hyperparameters in our model: number of neighbors in $k-$NN graphs, k (on the x-axis), and the entropic regularization coefficient, ϵ (on y-axis). The color of the scattered dots correspond to the average FOSCTTM values we receive for each alignment, with lower values corresponding to better alignments. The hyperparameter combinations that yielded the best FOSCTTM values are in black squares. 42

3.7 Ablation test results. We considered several modifications to algorithmic choices in SCOT and investigated the range of average FOSCTTM values we received in our alignments for scGEM (blue) and SNARE-seq (orange) datasets. The modifications considered are: (1) removing the entropic regularization term from the Gromov-Wasserstein optimal transport objective function, (2) using Euclidean distances for intra-domain distance and (3) using correlation-based distances instead of graph distances for the intra-domain distance matrices. 43

4.1 An example of the cell-type representation imbalance observed in real-world unpaired single-cell multi-omic datasets. This particular example is from scNMT-seq dataset. While this dataset is generated via a co-assaying technology (i.e. paired multi-omic dataset), the cell-type representation disproportion arises because different number of cells are retained after the quality control procedure is carried out for each measurement modality. 46

4.2 Overview of SCOTv2 on scNMT-seq dataset [20], which contains unbalanced cell-type representation across three domains - RNA expression, chromatin accessibility, and DNA methylation. SCOTv2 selects an anchor domain (denoted with *) and aligns other measurements to it. First, it computes intra-domain distances matrices D^m for $m = 1, 2, 3$, which are used to solve for correspondence matrices between the anchor and other domains. The circle sizes in the matrices depict the magnitude of the correspondence probabilities or how much mass to transport. Unbalanced GW relaxes the mass conservation constraint, so the transport map does not need to move each point with its original mass. Finally, it either co-embeds the domains into a common space or uses barycentric projections to project them onto the anchor domain. 47

4.3 Schematic visualizing the effect of the mass relaxation term in SCOTv2 (unbalanced Gromov-Wasserstein optimal transport). A. By allowing for the marginal distributions of the coupling matrix to diverge from the dataset marginals, we let the mass of each datapoint to be locally modified during transportation to better match cells from similar cell types, yielding better alignments for datasets with disproportionate cell-type representation. B. An example comparing SCOT and SCOTv2 alignments on SNARE-seq dataset alignment, with subsampled cell-type clusters in the chromatin accessibility domain to simulate cell-type imbalance. Notice that SCOT moves cells from overrepresented cell-types (e.g. BJ) in the place of underrepresented cell-types (e.g. K562), while SCOTv2 more correctly aligns cells. 57

4.6 Alignment results on simulations with co-assay datasets. **A** visualizes the alignment results by SCOTv2, using barycentric projection, on co-assay datasets SNARE-seq and scGEM when a cell-type is missing in the gene expression domain. **B** quantifies the alignment quality in this experiment by using the label transfer accuracy metric and compares to baseline methods. **C** plots the average label transfer accuracy results obtained from SCOTv2, SCOT, and Pamona algorithms when aligning randomly downsampled datasets. These experiments are repeated five times and the standard deviation is shown with error bars. 66

4.7 Runtimes for SCOTv2, SCOT, Pamona, UnionCom, and MMD-MA as the number of samples increases. 68

4.4 Alignment results for simulations and balanced co-assay datasets. **A** visualizes the barycentric projection alignment on SNARE-seq and scGEM for the full co-assay datasets, simulations with a missing cell-type in the epigenomic domain, and subsampled cell-types in both domains. **B** compares the alignment performance of SCOTv2 to the benchmarks through LTA. For SCOTvs, Pamona, and UnionCom, we report results on both embedding into a shared space (solid bars) and the barycentric projection (dotted bars). 70

4.5 Alignment results for multi-modal ($M > 2$) and separately sequenced datasets. **A** visualizes the alignment of scNMT-seq, sciOmics, and MEC. All datasets have unequal sample sizes and cell-type proportions across domains. **B** benchmarks alignment performance through LTA. As in Figure 4.4, we report results both by embedding (solid bars) and barycentric projection (dotted bars) for the methods that allow for both. For scNMT-seq and sciOmics, which are three-modal datasets, we only demonstrate results for SCOTv2, Pamona, and UnionCom, which can handle more than two modalities. 71

5.1 Schematic outlining the SCOOTR algorithm. Given two single-cell multi-omic datasets, SCOOTR seeks to find two coupling matrices, one aligning cells, and the other aligning features across these datasets. 74

5.2 Cell-cell and feature-feature alignment results on CITE-seq dataset. **A** visualizes the original domains – antibody abundance, and gene expression, respectively –, following dimensionality reduction with 2D principal component analysis (PCA). **B** visualizes the aligned domains, after the gene expression domain has been projected onto the antibody abundance domain via barycentric projection. **C** Feature alignment probabilities recovered by SCOOTR. The green boxes along the diagonal indicate the "ground-truth" correspondences we expect to see between the antibodies and their encoding genes. **D**. The feature alignment probabilities recovered by bindSC. 86

5.3 Cell and feature alignment results on SNARE-seq dataset. **A** visualizes the original domains – chromatin accessibility, and gene expression, respectively, following dimensionality reduction with 2D principal component analysis (PCA). **B** visualizes the aligned domains, after the chromatin accessibility domain has been projected onto the gene expression domain domain via barycentric projection. **C** Sankey plot visualizing the top chromatin accessibility feature correspondences recovered for the cell-type marker genes. These correspondences include the chromosomal regions of the marker genes and regions with predicted cell-type specific transcriptional factor (TF) binding. 87

5.4 **Cell-type alignment results on cross-species dataset.** Full supervision is provided on the $10,816$ paralogous genes between mice and lizard. 90

Chapter 1

Introduction

1.1 Background on Single-cell Multi-omics

All processes in cells –such as cell division, differentiation, heat shock response, enzyme/hormone secretion, signaling, DNA repair, etc.– are carried out by proteins and ribonucleic acids (RNAs). The blueprint for the production of these molecules are encoded in genes. The timing of their production, as well as their abundance, are determined by tightly regulated gene expression programs [21, 67, 68]. While differentially regulated gene expression programs can lead to the specialized cell types that carry out specific functions in an organism (e.g. motor neurons, skin cells etc), their misregulation or dysregulation lead to disorders [67]. As a result, studying the regulatory mechanisms behind gene expression is of interest for both fundamental and biomedical research.

1.1.1 Genome regulation and sequencing

Gene expression is regulated on multiple levels in the genome. In eukaryotic cells, the DNA is wrapped around histone proteins, forming the chromatin [21]. The chromatin is then further coiled and packaged into the nucleus (Figure 1.1). Genome architecture studies have revealed that the 3D structure that results from this coiling influences gene expression regulation as it determines which genomic regions will be

physically located close together in the nucleus, allowing or disallowing for protein binding events on the genome [21]. Reversible chemical modifications on the chromatin, such as acetylation, control the unwrapping of the DNA in a given region, making it accessible as a template for transcription. When the DNA is unwrapped, various proteins and RNAs can bind on these accessible regions, either driving gene expression initiation or preventing the binding of other proteins (e.g. RNA polymerase) to supress gene expression [21]. So, obtaining a mechanistic view of the gene expression regulation requires studying multiple properties of the genome. Thanks to the advances in molecular biology, we have various sequencing technologies to measure each of these features, such as HiC assays [78] to probe 3D structure of the genome, bisulfite sequencing techniques [60] to sequence the methylated regions of the genome, assays for transposase-accessible chromatin (ATAC) sequencing [10] to find accessible regions of the chromatin, chromatin immunoprecipitation (ChIP) assays [48] to identify protein binding events on the DNA, RNA-sequencing assays [96] to measure gene expression levels.

1.1.2 Single-cell sequencing and multi-omics

Until 2009, genome sequencing was only available at the bulk (i.e. cell population) level [101]. Laboratory scientists would lyse a culture of cells and purify the molecule of interest (e.g. mRNA or DNA) pooled in the culture. Sequencing of these molecules, then, yielded an *average* read-out for the whole cell population. In 2009, Tang *et al.* [87] performed the first single-cell (transcriptomic) sequencing experiment by isolating just one individual cell from a four-cell stage mouse blastomere. Then, Islam *et al.* in 2021 [47], and later Hashomshony in 2012 *et al.* [43], developed multiplexed single-cell transcriptomic sequencing methods, respectively called STRT-seq and CEL-seq, where they individually barcoded mRNA molecules from multiple isolated cells and then pooled them together in one sequencing run. These protocols scaled single-cell

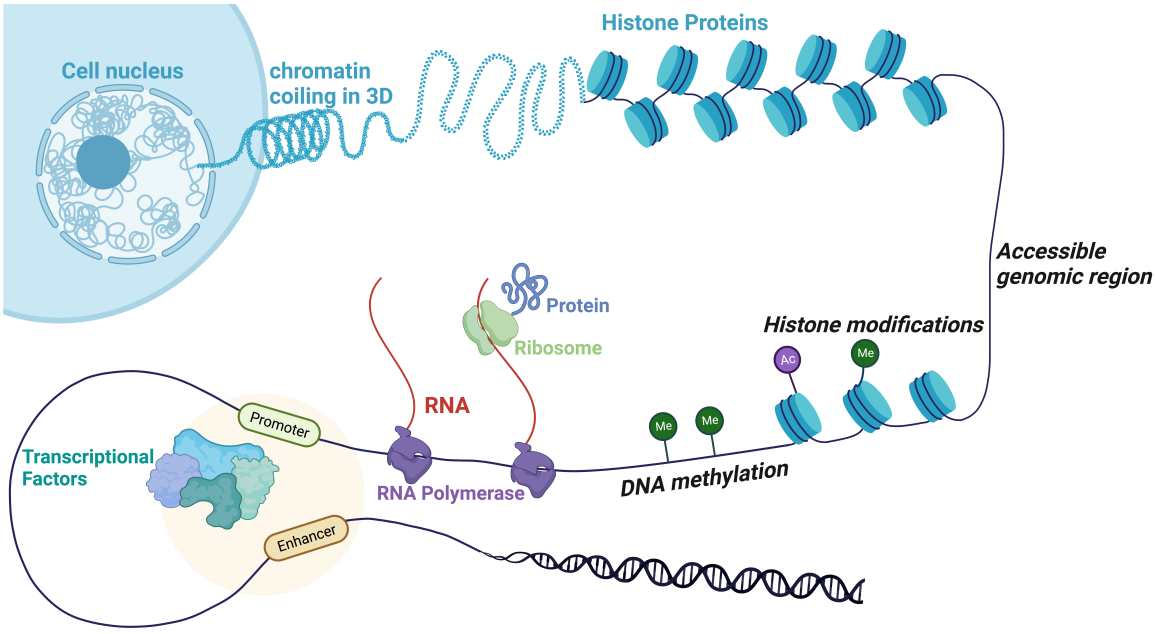

Figure 1.1: Overview of the eukaryotic genome architecture and some gene regulatory events. The figure is prepared on `Biorender.com`

sequencing to the order of $\sim$ 100 cells and showed that measurements taken at the single-cell resolution can reveal genomic heterogeneity in cell populations otherwise unrevealed by bulk sequencing. However, these protocols were still low-throughput, time- and labor-intensive. Droplet-based methods reported independently by Klein *et al* [51] and Macosko *et al* [59] in 2015 significantly increased the throughput of single-cell experiments by automating the isolation and barcoding of cells in droplets with microfluidic devices. Since then, private companies have improved sequencing protocols and commercialized these methods, making them accessible to a larger number of labs [101, 109]. Additionally, using similar approaches for cell isolation and barcoding, scientists have extended single-cell sequencing protocols to other types of genomic modalities [46], such as chromatin accessibility [11], DNA methylation [80], 3D conformation of the genome [72], protein binding events [75] etc.

Obtaining genomic measurements *at the single-cell resolution* is critical to gain a deeper understanding of genomic regulatory processes [67] Bulk sequencing provides

only one data point for an entire population of cells, which represents the average measurement for all cells in that population. However, single-cell sequencing yields an individual data point for each cell, revealing the entire distribution of biological states. By observing how the different genomic features co-vary across cells, researchers can determine which genomic features regulate which genes, and how these relationships change across different cell types or stages of cell differentiation [52, 67]. To study these relationships across various genomic features, researchers must take multi-modal (multiple types of) genomic measurements, which are called "multi-omic" measurements, on single cells [67].

Unfortunately, taking multi-omic measurements at the single-cell resolution is challenging due to the destructive nature of sequencing assays [6, 52, 57]. Most sequencing assays require lysing cells, which means cells will not be available for a subsequent sequencing experiment. This is not an issue for bulk-level sequencing, where researchers can aliquot a cell population into sub-populations and then apply a different sequencing assay on each [4]. As long as the sub-populations are large enough, their average measurements will reflect the average for the original cell population. This approach will not work when we want to obtain multi-omic measurements on the same single cells. However, for some combinations of genomic measurements (Figure 1.2), researchers have developed experimental protocols to obtain multi-omic measurements from single-cells. These protocols, hereinafter called “co-assays”, follow one of three strategies [4, 82]:

1. **Applying a non-destructive assay (e.g. one based on fluorescent reporters), followed by a destructive one (e.g. sequencing):** This strategy is typically used in plate-based single-cell sequencing protocols that rely on florescence-activated cell sorting (FACS) to sort individual cells into microtitre plates. In this strategy, FACS index sorting allows for gathering cytometric data (e.g. cell surface markers) about the cell before performing sequencing.

Using this strategy, for example. Wilson *et al.* have identified cell surface markers associated with haematopoietic stem cells while performing single-cell RNA sequencing [99].

- **Limitations:** The possible cellular properties that can be revealed by this approach is limited to the ones that can be profiled by fluorescent reporters, as well as by the spectral overlap between these reporters [82] Additionally, plate-based sequencing technologies that use FACS is typically lower throughput than the droplet-based methods [82, 101].

2. **Isolation of different types of molecules (e.g. DNA and RNA) from the same cell upon lysis, then separately sequencing them:** Several groups have achieved joint genomic and transcriptomic profiling of single-cells by physically separating cell nucleus from the cytoplasm or isolating mRNA molecules using biotinylated or paramagnetic oligo(dT) beads, then independently sequencing the mRNAs and the DNA. Examples include G&T-seq [58], which yields gene expression measurements along with DNA copy number variation and genome sequencing data, scMT-seq [3] and scM&T-seq [45], both of which reveal DNA methylation status jointly with gene expression measurements, as well as scTrio-seq [44], which is a tri-modal sequencing method giving information on DNA methylation, gene expression and DNA copy number variations together.

   - **Limitations:** This approach is limited to the profiling of cellular properties that can be detected through different types of molecules. Additionally, the quality of data gathered through these co-assays is typically lower than unimodal single-cell sequencing experiments, as explained below [36, 46, 82].

3. **Conversion of multiple cellular features to a common molecular format that can be sequenced in one run:** Recently developed CITE-seq [81] and

REAP-seq [100] assays use this method to jointly profile cell surface proteins (i.e. "epitopes") and mRNA levels in one sequencing run. They barcode surface proteins with oligonucleotides using the antibodies that target them. They then sequence these oligonucleotides jointly with the cDNAs synthesized from cellular mRNA. Since this approach is compatible with droplet-based sequencing protocols, they have higher throughput than fluoresent reporter-based methods [82]

- **Limitations:** While this has been a scalable and promising approach, sequencing of features that require access to the same part of the genome without interfering with the measurements of each other remains to be a challenge.

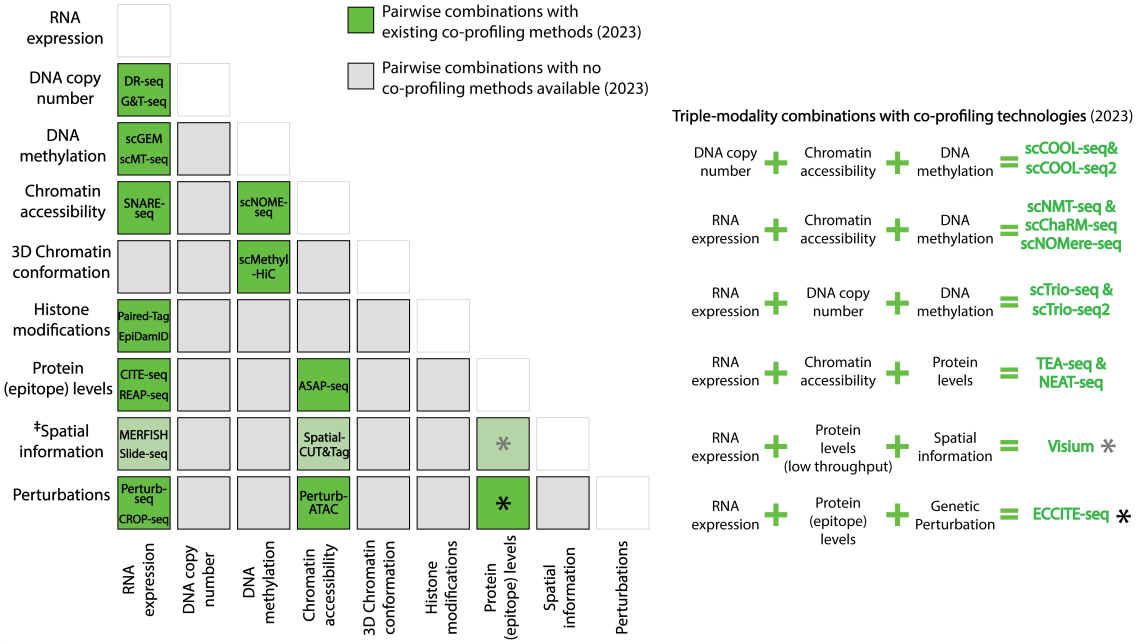

Figure 1.2: A chart of genomic and cellular feature combinations for which at least one single-cell co-assaying technology is available (shown in green). For each combination, we give a *non-exhaustive* list of example protocols [16–20, 24, 32–34, 40, 41, 44, 45, 54, 58, 63, 64, 71, 73, 74, 76, 81, 85, 95, 100, 102, 103, 110]. This figure is adopted from [57] and updated with information from [91] to include new assaying protocols available in 2023.

These three approaches, as outlined above, have enabled the development of a growing number of single-cell co-assaying protocols. From these protocols, a few have been commercialized (such as 10x Chromium Single-cell Multiome ATAC + Gene Expression sequencing [39] or Visium [40]). However, as Figure 1.2 demonstrates, there are no co-assays available for majority of the combinations of cellular and genomic features [57]. Each new combination requires the development of a new experimental protocol, and the number of possible combinations grows combinatorially as we start to consider possibilities beyond pairwise. Moreover, developing co-assaying protocols is especially difficult when multiple measurements need access to the same part of the genome without interfering with one another [46]. Even without this challenge, the data yielded from co-assays tend to be noisier than data obtained from the uni-modal single-cell profiling experiments [4, 36, 46]. One reason behind this is that different sample treatment procedures might be ideal for different sequencing modalities [46].

In settings where there are no co-assays available for the measurement combination of interest, or co-assaying is not preferable due to the data quality concerns or budget constraints, scientists approach single-cell sequencing in a manner similar to bulk sequencing: They divide a cell population into aliquots and use each aliquot for a different single-cell sequencing experiment. Then, integrating the data from these experiments to obtain a joint view requires computational approaches [52].

1.2 Thesis Overview and Summary of Contributions

This thesis introduces three computational methods for integrated analysis of separately profiled (i.e. unpaired) single-cell multi-omic datasets. The algorithms we present heavily rely on the "optimal transport theory", which is a mathematical framework that relates probability measures to one another. We first introduce optimal transport theory in Chapter 2, then detail three algorithms that we developed. The algorithms include:

1. **In Chapter 3, Single-cell alignment with optimal transport (SCOT)**: An unsupervised algorithm that probabilistically aligns cells from two unpaired single-cell datasets generated through separate sequencing experiments. The algorithm performs alignment by comparing the underlying dataset geometries using Gromov-Wasserstein optimal transport [61]. We additionally develop a heuristic for automatically self-tuning hyperparameters for cases where users may not have sufficient validation data to for selecting hyperparameters, which is a common real-world scenario. This is a unique feature of SCOT compared to the other unsupervised single-cell multi-omic alignment methods (as reviewed in Chapter 3). We demonstrate through experiments with simulated and real-world single-cell multi-omic datasets that when validation data is available for hyperparameter tuning, SCOT performs on par with the state-of-the-art single-cell multi-omic alignment algorithms that exist since the time of its development. However, when the users lack validation data, its self-tuning procedure provides a significant advantage over these algorithms. This work was presented at the 15^{th} *Machine Learning in Computational Biology* conference, 37^{th} *International Conference on Machine Learning (ICML) Workshop on Computational Biology (WCB)*, and the 25^{th} *International Conference on Research in Computational Molecular Biology (RECOMB)*, then subsequently published in the 2021 RECOMB special issue of the *Journal of Computational Biology* [28].

2. **In Chapter 4, SCOTv2:** An extension of the SCOT algorithm that handles datasets with disproportionate cell type representation and also integrates more than two datasets together. We show through experiments with real-world datasets that cell-type proportion disparities is a realistic scenario in real-world sequencing experiments and that most of the existing single-cell multi-omic integration algorithms fail to yield quality alignments in this case. This work was presented at the 16^{th} *Machine Learning in Computational Biology* conference

and 26^{th} *International Conference on Research in Computational Molecular Biology (RECOMB)* [29], then published in the 2022 RECOMB special issue of the *Journal of Computational Biology* [27].

3. **In Chapter 5, Single-cell fused gromov co-optimal transport (SGCOOTR)**: A single-cell multi-omic alignment method that jointly and iteratively aligns both samples and features of input datasets based on a novel optimal transport formulation we propose. This formulation is based on an interpolation between the Gromov-Wasserstein distance [61] and co-optimal transport [88]. In the sample alignment step, SGCOOTR leverages information on both the structure of the dataset through pairwise distances between samples and the feature-feature relationships across datasets. In the feature alignment step, SGCOOTR performs optimal transport between the features of the *transformed* sample space. We demonstrate through experiments with both single-cell sequencing datasets and established machine learning benchmarks that the proposed interpolation improves upon sample alignments (i.e. cell-cell in single-cell datasets) by leveraging the additional information between features compared to SCOT and SCOTV2 and also by interpolating the different transformation invariance properties of the Gromov Wasserstein distance and co-optimal transport. Our experiments with single-cell datasets show that the proposed method can be used as a hypothesis generation tool for feature relationships and allows for providing partial supervision on either the feature- or the cell-level alignments in order to improve the alignment quality of both. An earlier formulation of this method was presented at the 36^{th} *Conference on Neural Information Processing (NeurIPS) Workshop on Learning Meaningful Representations of Life (LMRL)*, as well as the 17^{th} *Machine Learning in Computational Biology (MLCB)* conference.

Chapter 2

Background on Optimal Transport

All algorithms presented in this thesis (summarized in Chapter 1.2) heavily rely on the optimal transport theory. Optimal transport is a mathematical framework that relates probability distributions or discrete measures to one another [70, 93]. Since the problem put forward in Chapter 1.1.2 is concerned with relating datasets from different single-cell measurements, which can be treated as empirical measures (see subsection 2.1), optimal transport lends itself to be a natural choice of approach for this problem. Here we give a brief overview of optimal transport before describing the algorithms we developed for single-cell multi-omic data integration tasks.

2.1 Definitions and Notations

- $\mathbb{X}, \mathbb{Y}$: Denote sets. For example $\mathbb{R}_+^d$ denotes d-dimensional set of positive real numbers
- $\mathcal{X}, \mathcal{Y}$: When the sets correspond to spaces, they are denoted by calligraphic uppercase letters
- $\mathcal{P}(\mathcal{X}), \mathcal{P}(\mathcal{Y})$: Denote the set of probability distributions defined on metric spaces $\mathcal{X}, \mathcal{Y}$
- $\mathbf{x}, \mathbf{y}$: Vectors are denoted by bold lowercase letters

- $\mathbf{X}, \mathbf{Y}$: Matrices are denoted by bold uppercase letters
- $[\![n]\!]$: Denotes set of all positive integers up to n, i.e. $\{1, 2, ..., n\}$
- $\mathbb{1}_n$: Denotes a vector of 1's of length n
- μ, ν: Denote continuous probability distributions, defined on spaces $\mathcal{X}, \mathcal{Y}$
- Discrete probability measures, $\mathbf{p}, \mathbf{q}$ are expressed as histograms with a vector of probabilistic weights defined at specific locations,

$$\mathbf{p} = \sum_{i=1}^{n} \boldsymbol{a}_i \delta_{\mathbf{x}_i} \tag{2.1}$$

$$\Sigma_n := \{\boldsymbol{a} \in \mathbb{R}_+^n : \sum_{i}^{n} \boldsymbol{a}_i = 1\} \tag{2.2}$$

  (i.e. a histogram of n bins, where each bin is represented as a point mass $\delta_{\mathbf{x_i}}$ with a magnitude of a_i), where:

  - Σ_n denotes the probability simplex the weights are sampled from,
  - $\delta_{\mathbf{x}_i}$ denotes a Dirac measure at position $\mathbf{x}_i$ (i.e. a unit of mass sharply and infinitely concentrated at location $\mathbf{x}_i$)

- $T_{\#} : \mathcal{P}(\mathcal{X}) \to \mathcal{P}(\mathcal{Y})$ denotes the pushforward operator, i.e. $T_{\#}\mu = \nu$ for $\mu \in \mathcal{P}(\mathcal{X})$ and $\nu \in \mathcal{P}(\mathcal{Y})$ (further described below in Section 2.2).
- $\boldsymbol{\Gamma}, \gamma$: Respectively denotes the discrete and continuous Kantorovich couplings (described below in Section 2.2)
- $\Pi(\mathbf{p}, \mathbf{q})$ or $\Pi(\mu, \nu)$: Denotes a set of admissible coupling matrices whose marginals (row distributions and column distributions) obey the probability distributions in transport (i.e. couplings that fully transport the input probabilities)
- $\langle A, B \rangle$: Denote the Frobenius inner product, i.e. $\sum_{ij} [\mathbf{A}]_{ij} [\mathbf{B}]_{ij}$

- $\exp(x)$ refers to the exponentiation of x, i.e. e^x

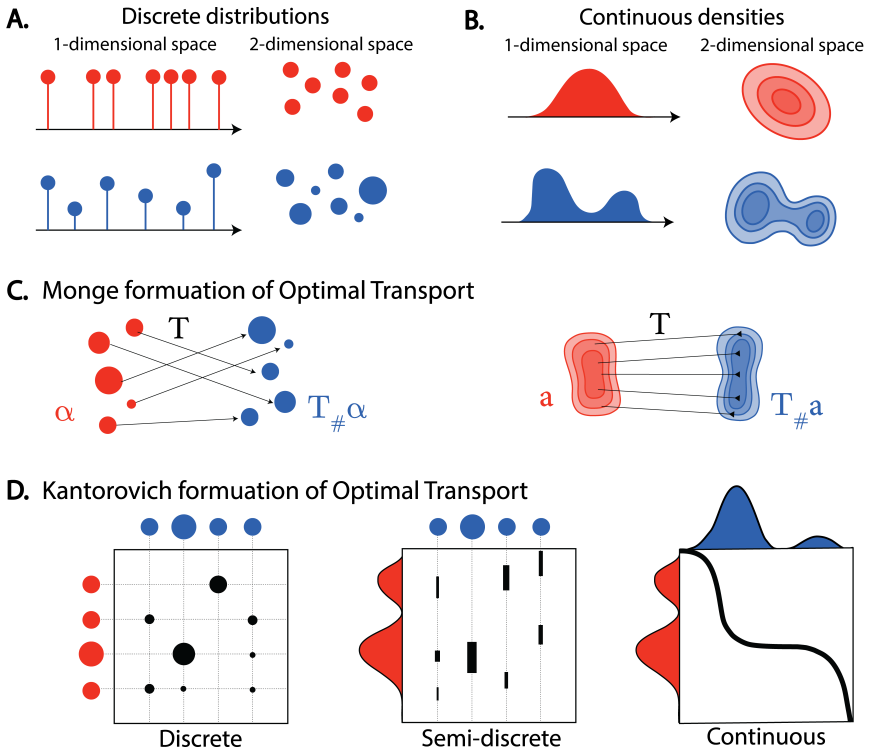

Figure 2.1: Schematic of optimal transport maps in Monge's and Kantorovich's formulations. Optimal transport relates probabily distributions, either defined as (A) discrete measures or (B) continuous densities. (C) The Monge formulation seeks to find deterministic pushforward maps that will transport probability distributions onto each other. (D) Kantorovich's formulation relaxes the transport problem and allows for probabilistic transport maps. This figure is adapted from Peyre and Cuturi [70].

2.2 Overview: The Monge Problem and the Kantorovich Relaxation

The Monge problem The origins of optimal transport theory can be traced back to the French mathematician and geometer, Gaspard Monge, from the 18^{th} century [93]. Monge studied transportation theory to enable efficient movement of resources for building military constructions. The problem he considered can be summarized as follows: Assume we have a certain amount of soil extracted from a the ground and we need to transport it to somewhere in the construction to be used. Where in the

construction should one send the soil from a specific location in order to minimize the total effort [93]? He generalized this problem to probability measures: Given two metric spaces $\mathcal{X}$ and $\mathcal{Y}$ and probability distributions from these spaces $\mu \in \mathcal{P}(\mathcal{X})$ and $\nu \in \mathcal{P}(\mathcal{Y})$, what transport map, $T : \mathcal{X} \to \mathcal{Y}$, could transfer all the probability mass of μ onto ν such that the overall cost of mass transfer, $c : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+ \cup +\infty$, would be minimized? Monge defined this transport map to be a deterministic *pushforward* operator, $T_{\#} : \mathcal{P}(\mathcal{X}) \to \mathcal{P}(\mathcal{Y})$, that is both a *surjective* and an *injective* map. With this pushforward operator, the Monge problem can be expressed as:

$$\inf_{T}\{\int_{\mathcal{X}} c(x, T(x))d\mu(x) | T_{\#}\mu = \nu\}$$

In the case of relating discrete measures, the pushforward operator is equivalent to finding a permutation that fully matches the point masses / histogram bins:

$$\min_{\sigma \in \mathrm{Perm}(n)} \frac{1}{n} \sum_{i=1}^{n} \mathbf{C}_{\mathbf{i},\sigma(\mathbf{i})}$$

Kantorovich relaxation In many practical applications of optimal transport, the deterministic pushforward operator described in Monge's assignment problem is too restrictive. For example, in the case of discrete measures, it cannot be used to compare histograms of different sizes, because a valid solution only exists if the two histograms have the same number of bins, as well as "compatible" masses (probabilities) defined over these bins (i.e. for each bin in one histogram, there is a bin with the same mass in the other histogram) [70]. Moreover, even in cases where a valid solution exists, computing the solution is too costly due to the combinatorial nature of the problem. In 1942, the Russian mathematician and economist, Leonid Kantorovich, proposed to relax Monge's assignment problem by allowing for probabilistic alignments [50]. In the case of discrete measures, this allows for *splitting* the mass of a bin onto multiple bins during alignment (Figure 2.1C). Kantorovich's formulation of optimal transport

seeks to find a ***coupling*** (a.k.a "correspondence measure"), $\mathbf{\Gamma}$, such that:

$$\min_{\Gamma \in \Pi(\boldsymbol{p},\boldsymbol{q})} \langle \boldsymbol{C}, \mathbf{\Gamma} \rangle = \min_{\Gamma \in \Pi(\boldsymbol{p},\boldsymbol{q})} \sum_{i}^{n_1} \sum_{j}^{n_2} \boldsymbol{C}_{i,j} \mathbf{\Gamma}_{i,j} \tag{2.3}$$

for discrete measures, with the linear constraints defined over the coupling matrix Γ as:

$$\Pi(\boldsymbol{p}, \boldsymbol{q}) = \{\mathbf{\Gamma} \in \mathbb{R}_{+}^{n_1 \times n_2} : \mathbf{\Gamma} \mathbb{1}_{n_2} = \boldsymbol{p},\ \mathbf{\Gamma}^T \mathbb{1}_{n_1} = \boldsymbol{q}\} \tag{2.4}$$

Here, each entry in the coupling $\mathbf{\Gamma}_{i,j}$ describes how much mass is split between the i^{th} bin from the first histogram, p (of length n_1), and the j^{th} bin from the second histogram, q(of length n_2), representing their correspondence probability. The transport cost $\boldsymbol{C}$ can be represented as a matrix in the discrete case, with each entry $\boldsymbol{C}_{i,j} = c(x^{(i)}, y^{(j)})$ giving the cost of alignment between the bins $i \in [\![n_1]\!]$ and $j \in [\![n_2]\!]$. The linear constraints in Equation 2.4 ensure that the coupling comes from a set of admissible measures $\Pi(\boldsymbol{p}, \boldsymbol{q})$, so that the histograms are transported in full and the marginals of the coupling matrix preserve $\boldsymbol{p}$ and $\boldsymbol{q}$.

The problem in Equation 2.3 can be generalized to continuous distributions as:

$$\min_{\gamma \in \Pi(\mu,\nu)} \int_{\mathcal{X} \times \mathcal{Y}} c(x, y) d\gamma(x, y) \tag{2.5}$$

such that:

$$\Pi(\mu, \nu) = \{\gamma \in \mathcal{P}(\mathcal{X} \times \mathcal{Y}) | P_{\mathcal{X}\#}\gamma = \mu, P_{\mathcal{Y}\#}\gamma = \nu\} \tag{2.6}$$

Hereinafter, we refer to the Kantorovich formulation when discussing optimal transport. Moreover, we will only discuss the discrete case since we only work with discrete measures defined on single-cell sequencing datasets in our applications.

2.3 Efficient Computational Solvers and Associated Regularizers

Although Kantorovich relaxation improves the computational complexity of the optimal transport problem by allowing for splitting probabilistic masses in transport, the recovered coupling matrices will be sparse in most cases due to the "least effort" principle of optimal transport. Sparse coupling may not always be ideal in practical applications and more dense couplings will make the problem more convex. To this end, an entropic regularization term over the coupling matrix is added in many optimal transport applications [70]:

$$\min_{\Gamma \in \Pi(\boldsymbol{p},\boldsymbol{q})} \sum_i \sum_j \boldsymbol{\Gamma}_{i,j} \boldsymbol{C}_{i,j} - \epsilon H(\boldsymbol{\Gamma}) \tag{2.7}$$

such that

$$\Pi(\boldsymbol{p},\boldsymbol{q}) = \{\boldsymbol{\Gamma} \in \mathbb{R}_+^{n_1 \times n_2} : \boldsymbol{\Gamma} \mathbb{1}_{n_2} = \boldsymbol{p},\ \boldsymbol{\Gamma}^T \mathbb{1}_{n_1} = \boldsymbol{q}\}$$

where $H(\boldsymbol{\Gamma}) = -\sum_i \sum_j \boldsymbol{\Gamma}_{ij} \log(\boldsymbol{\Gamma}_{ij})$ is the Shannon entropy.

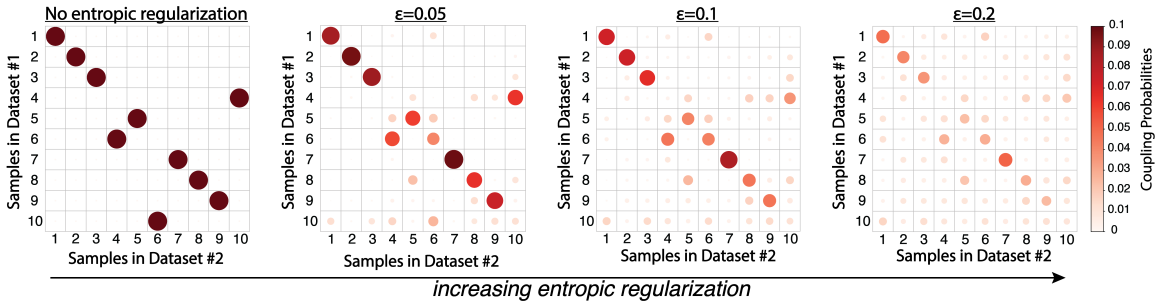

Figure 2.2: **Visualizing the effect of entropic regularization on the optimal coupling.** As the entropic regularization coefficient increases, the coupling probabilities are more split, yielding a less sparse solution. In this example, we align two datasets with 10 MNIST handwritten digit samples [53] in each, as a toy dataset.

In practice, the coefficient of the entropic regularization term, ϵ controls the extent of sparsity in the coupling, as demonstrated in Figure 2.2 above, and this regularization yields an efficient computation procedure to find the optimal coupling, as derived

below.

The Lagrangian of the function in Equation 2.7 is:

$$L(\mathbf{\Gamma}, \lambda^{(1)}, \lambda^{(2)}) = \underbrace{\sum_i \sum_j \mathbf{\Gamma}_{i,j} \boldsymbol{C}_{i,j} + \epsilon \sum_i \sum_j \mathbf{\Gamma}_{ij} \log(\mathbf{\Gamma}_{ij})}_{\text{Primary problem}} - \underbrace{\lambda^{(1)^T}(\mathbf{\Gamma}\mathbb{1} - p) - \lambda^{(2)^T}(\mathbf{\Gamma}^T\mathbb{1} - q)}_{\text{Linear constraints}} \tag{2.8}$$

At optimality, we will have:

$$\frac{\partial L}{\partial \mathbf{\Gamma}_{i,j}} = 0 = \boldsymbol{C}_{ij} + \epsilon \log(\mathbf{\Gamma}_{ij}) - \boldsymbol{\lambda}_i^{(1)} - \boldsymbol{\lambda}_j^{(2)} \tag{2.9}$$

$$\implies \epsilon \log(\mathbf{\Gamma}_{ij}) = -C_{ij} + \boldsymbol{\lambda}_i^{(1)} + \boldsymbol{\lambda}_j^{(2)} \tag{2.10}$$

$$\implies \log(\mathbf{\Gamma}_{ij}) = \frac{-\boldsymbol{C}_{ij} + \boldsymbol{\lambda}_i^{(1)} + \boldsymbol{\lambda}_j^{(2)}}{\epsilon} \tag{2.11}$$

$$\implies \mathbf{\Gamma}_{ij} = \exp\left(\frac{-\boldsymbol{C}_{ij} + \boldsymbol{\lambda}_i^{(1)} + \boldsymbol{\lambda}_j^{(2)}}{\epsilon}\right) \tag{2.12}$$

$$\implies \mathbf{\Gamma}_{ij} = \exp\left(\frac{\boldsymbol{\lambda}_i^{(1)}}{\epsilon}\right) \exp\left(\frac{-\boldsymbol{C}_{ij}}{\epsilon}\right) \exp\left(\frac{\boldsymbol{\lambda}_j^{(2)}}{\epsilon}\right) \tag{2.13}$$

In Equation 2.13, the expression $\exp\left(\frac{-C_{ij}}{\epsilon}\right)$ is a constant. In fact, it is the Gibbs kernel associated with the cost matrix $\boldsymbol{C}$, which does not change. We will denote this by the matrix $\mathbf{K}$. The other two expressions are scaling vectors based on the unknown optimal Lagrangian multiplies $\lambda^{(1)}$ and $\lambda^{(2)}$. Let:

$$\mathbf{u} = \exp\left(\frac{\boldsymbol{\lambda}^{(1)}}{\epsilon}\right) \tag{2.14}$$

$$\mathbf{v} = \exp\left(\frac{\boldsymbol{\lambda}^{(2)}}{\epsilon}\right) \tag{2.15}$$

$$\text{such that } [\text{Diag}(\mathbf{u})\mathbf{K}\text{Diag}(\mathbf{v})]\mathbb{1} = \mathbf{p} = \mathbf{\Gamma}\mathbb{1} \text{ and} \tag{2.16}$$

$$[\text{Diag}(\mathbf{v})\mathbf{K}^\mathbf{T}\text{Diag}(\mathbf{u})]\mathbb{1} = \mathbf{q} = \mathbf{\Gamma}^\mathbf{T}\mathbb{1} \tag{2.17}$$

where $\text{Diag}(\mathbf{u})$ denotes a matrix with the elements of the vector $\mathbf{u}$ in its diagonal entries, with 0s elsewhere. Note that the product of a diagonal matrix with $\mathbb{1}$ yields a colun vector containing the diagonal entries from the matrix. So, we can simplify

the expression in Equation 2.17 with:

$$\text{Diag}(\mathbf{u})\mathbf{K}\mathbf{v} = \mathbf{p} \tag{2.18}$$

$$\text{Diag}(\mathbf{v})\mathbf{K}^{\mathbf{T}}\mathbf{u} = \mathbf{q} \tag{2.19}$$

This system of equations can be solved by the following algorithm:

Algorithm 1: Sinkhorn iterations

Given the entropic regularization coefficient, ϵ, and the cost matrix $\mathbf{C}$

Compute $\mathbf{K}=\frac{-\mathbf{C}}{\epsilon}$

Initialize $\mathbf{v^{(0)}} = [1, 1, ..., 1]^T$

Until convergence:

    Update $\mathbf{u}^{(t)}$ so that Equation 2.18 holds:

        $\mathbf{u}^{(t)}=\frac{\mathbf{p}}{\mathbf{K}\mathbf{v}^{(\mathbf{t}-\mathbf{1})}}$

    Update $\mathbf{v}^{(t)}$ so that Equation 2.19 holds:

        $\mathbf{v}^{(t)}=\frac{\mathbf{q}}{\mathbf{K}^{\mathbf{T}}\mathbf{u}^{(\mathbf{t})}}$

After convergence (let's say, reached after n iterations):

Return: $\mathbf{\Gamma} = \text{Diag}(\mathbf{u}^{(\mathbf{n})})\mathbf{K}\ \text{Diag}(\mathbf{v}^{(\mathbf{n})})$

There are other regularizers and solvers developed for the efficient computation of the Kantorovich optimal transport problem. For more information on these, we refer the readers to [70]. In the subsequent algorithms presented in this thesis, we employ entropic regularization and mainly rely on the Sinkhorn iterations presented in Algorithm 10, which we simply refer to as "Sinkhorn()" function from now on.

Chapter 3

SCOT: Unsupervised Single-cell alignment with (Gromov-Wasserstein) Optimal Transport

3.1 Introduction

As introduced in Chapter 1.1, the growing variety of single-cell assays allows us to measure the heterogeneous landscape of cell state in a sample, revealing distinct subpopulations and their developmental and regulatory trajectories across time. Different technologies can interrogate different molecular aspects of the cell, such as gene expression, protein synthesis, chromatin accessibility, DNA methylation, histone modifications, and chromatin 3D confirmation. Combining data generated by these single-cell assays can provide novel insights into the interactions between these molecular views and their joint regulatory mechanisms. Hence, learning this combined information is critical to our understanding of complex biological processes and heterogeneous diseases. Despite its importance, combining single-cell multi-omics data is a challenging task. Aside from some co-assay procedures that simultaneously isolate separate molecular material for each measurement, applying multiple assays on the same single cell is not currently possible (more information in Chapter 1.1.2) In such

cases, the measurements are taken by dividing a cell population into subpopulations and assaying them separately, losing the potential for 1–1 correspondence of cells that is required for easy data integration (Figure 3.1).

In recent years, computational methods have been developed to solve the single-cell data integration problem. Many of these methods combine different experiments from a single modality such as RNA sequencing for correcting batch effects [2, 7, 83, 97, 98]. However, integrating data from multiple modalities such as gene expression and DNA methylation presents unique challenges. For example, when we measure different properties of a cell, we cannot *a priori* identify correspondences between features in the two domains. Accordingly, integrating two or more single-cell data modalities requires methods that rely on neither common cells nor features across the data types. This aspect prevents the application of some existing single-cell alignment methods to unsupervised settings because they require some correspondence information to perform alignment [2, 7, 83, 97, 98]. Earlier versions of the popular batch integration method Seurat required correspondence information in the form of cells from a similar biological state that are shared across the two datasets (known as "anchor points").While a more recent version automatically selects these anchor points, it still requires features from one domain to be mapped to the other domain to perform the single-cell alignment [83]. This mapping might be possible for experiments like gene expression and chromatin accessibility, where one can map the chromatin region read counts to the corresponding gene regions. However, it can be difficult to perform for other sequencing assay combinations. Furthermore, [13] have shown that such methods do not yield quality alignments in unsupervised settings.

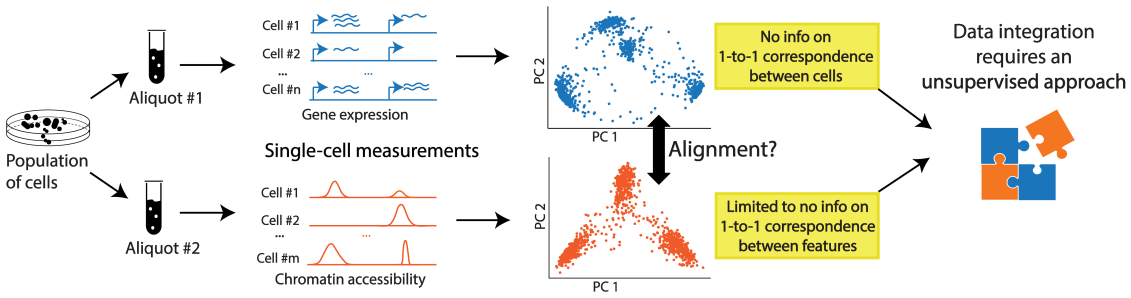

Figure 3.1: **Visualization of the single-cell multi-omic integration problem.** When experimentally co-profiling different aspects of the genome on single-cells is not possible, scientist apply different single-cell sequencing methods on separate aliquots of a cell population. This procedure yields disparate datasets as plotted, with limited to no prior information on 1-1 correspondences either between cells or features.

3.1.1 Existing Approaches (as of the acceptance for publication in 2020)

Multiple approaches have tried to align datasets in an entirely unsupervised fashion. One of the earliest attempts, the joint Laplacian manifold alignment (JLMA) algorithm, constructs eigenvector projections based on k-nearest neighbor graph Laplacians of the data [94]. The generalized unsupervised manifold alignment (GUMA) [22] algorithm seeks a 1–1 correspondence between two datasets based on optimization of a local geometry matching term. Liu *et al.* [56] showed that these methods do not perform well on the single-cell alignment task and proposed an alignment algorithm based on the maximum mean discrepancy (MMD) measure, called MMD-MA. Another method, UnionCom [13], extends GUMA to perform unsupervised topological alignment and makes it more suitable for single-cell multi-omics integration. While MMD-MA aims to match the global distributions of the datasets in a shared latent space, UnionCom emphasizes learning both local and global alignments between the two distributions. Neither method requires any correspondence information, either among samples or features, to perform an alignment. The respective papers demonstrate state-of-the-art performance on simulated and real datasets. Although these results are encouraging, MMD-MA and UnionCom require that the user specify three

and four hyperparameters, respectively. Hyperparameter selection can significantly affect the quality of alignments. Therefore, in an unsupervised real-world setting with no validation data on correspondences, hyperparameter tuning can be difficult to perform and can lead to sub-par alignments.

3.1.2 Our contributions

In this chapter, we propose an unsupervised alignment method based on optimal transport theory. Optimal transport finds the most cost-effective way to move data points from one domain to another. One way to think about it is as the problem of moving a pile of sand to fill in a hole through the least amount of work. Traditionally, optimal transport problems have been difficult to compute, especially for large-scale datasets. However, subsequent relaxations [50, 70] modify the original optimal transport problem, making it more applicable and easier to compute. Recently, several regularization procedures [69] have further improved the computational scalability of optimal transport.

In biology, an emerging number of applications are using optimal transport to learn a mapping between data distributions [1, 12, 77, 104, 105]. Schiebinger *et al.* [77] use it to study temporal changes in gene expression by using regularized unbalanced optimal transport to compute expression differences between time points. SpaOTsc [12] maps cells with high ligand expression onto cells with high receptor expression to recover cell signaling relationships in spatially resolved single-cell RNA-seq datasets. ImageAEOT [105] maps single-cell images to a common latent space through an autoencoder and then uses optimal transport to track cell trajectories. In related work, the same authors use autoencoders and optimal transport to learn transport maps among multiple domains [104]. However, the application of their method to single-cell datasets requires some form of supervision, like class labels, to be used during transport.

The classic optimal transport problem requires datasets from the same metric space. Mémoli *et al.* [62] generalizes optimal transport to the Gromov-Wasserstein distance, which compares metric spaces directly instead of comparing samples across spaces, making optimal transport suitable for multi-modal alignment. In natural language processing, Alvarez *et al.* [1] use this approach to measure similarities between pairs of words across languages to compute the similarity between languages. As far as we are aware, the only biological application of Gromov-Wasserstein optimal transporta comes from [66], which uses it to reconstruct the spatial organization of cells from transcriptional profiles.

We present Single-Cell alignment using Optimal Transport (SCOT), an unsupervised algorithm that uses Gromov-Wasserstein-based optimal transport to align single-cell multi-omics datasets (presented schematically in Figure 3.2). Like UnionCom, SCOT aims to preserve local geometry when aligning single-cell data. SCOT achieves this by constructing a k-nearest neighbor ($k-$NN) graph for each dataset (or domain) and then computing graph distance matrices for each $k-$NN graph to capture the intra-domain distances. SCOT then finds a probabilistic coupling matrix that minimizes the discrepancy between the intra-domain distance matrices. Finally, it uses the coupling matrix to project one single-cell dataset onto another through barycentric projection, thus aligning them. Unlike MMD-MA and UnionCom, SCOT requires tuning only two hyperparameters and is robust to the choice of one. We compare the alignment performance of SCOT with MMD-MA and UnionCom on four simulated and two real-world datasets. SCOT aligns datasets as well as the state-of-the-art methods and scales well with increasing numbers of samples. Moreover, we demonstrate that the Gromov-Wasserstein distance can guide SCOT's hyperparameter tuning in a fully unsupervised setting when no orthogonal alignment information is available. Thus, unlike other methods, SCOT provides a heuristic for hyperparameter selection without validation data.

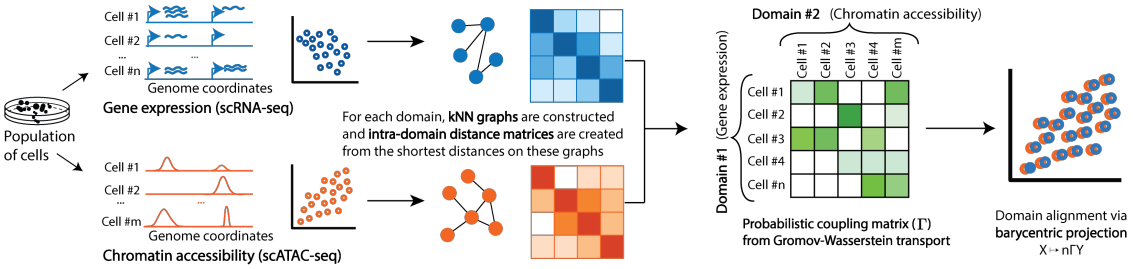

Figure 3.2: **Schematic of SCOT alignment of single-cell multi-omics data.** A population of cells is aliquoted for different single-cell sequencing assays. SCOT constructs k-NN graphs based on sample-wise correlations and finds a probabilistic coupling between the samples of each domain that minimizes the distance between the two intra-domain graph distance matrices. Barycentric projection projects one domain onto another based on this coupling matrix.

3.2 Methods

SCOT relies on Gromov-Wasserstein optimal transport to move data points from one domain to another while preserving the original local geometry. The goal of the transport problem at the core of SCOT is to find an ideal "coupling" (also called "correspondence") matrix that describes the probability of alignment between each point across domains. In this section, we first briefly re-introduce optimal transport theory, followed by its extension to Gromov-Wasserstein distance. Then, we present the details of our algorithm.

We have two datasets representing two domains, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n_x})$ from $\mathcal{X}$ and $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{n_y})$ from $\mathcal{Y}$. The datasets have n_x and n_y points, respectively. We do not require any correspondence information or assume that there is any ground truth for 1—1 correspondence between samples or features, but we do assume there is some underlying shared biology (e.g. cells across the datasets sharing a lineage or belonging to shared cell types), so that the datasets can be meaningfully aligned.

3.2.1 Optimal Transport

The Kantorovich optimal transport problem seeks to find a minimal cost mapping between two probability distributions or discrete measures [70]. Referring back to the problem of moving a sand pile to fill in a hole, Kantorovich optimal transport allows us to split the mass of a grain of sand instead of moving the whole grain; therefore, the mappings need not be 1—1. Consider discrete measures $\mathbf{p}$ and $\mathbf{q}$ as such

$$\mathbf{p} = \sum_{i=1}^{n_x} a_i \delta_{\mathbf{x}_i} \text{ and } \mathbf{q} = \sum_{j=1}^{n_y} b_j \delta_{\mathbf{y}_j},$$

where $\sum_{i=1}^{n_x} a_i = 1 = \sum_{j=1}^{n_y} b_j, a_i \geq 0, b_j \geq 0$ and $\delta_{\mathbf{x}_i}$ is the Dirac measure. This optimal transport problem finds a minimal coupling Γ that attains

$$\min_{\Gamma \in \Pi(\mathbf{p},\mathbf{q})} \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} c(\mathbf{x}_i, \mathbf{y}_j) d\Gamma_{i,j} \tag{3.1}$$

$$\text{subject to: } \Gamma(i,j) \geq 0, \sum_{i=1}^{n_x} \Gamma_{i,j} = \mathbf{q}_j, \sum_{j=1}^{n_y} \Gamma_{i,j} = \mathbf{p}_i$$

where $c(\mathbf{x}_i, \mathbf{y}_j)$ is a cost function defined over the samples from the two datasets and $\Pi(\mathbf{p}, \mathbf{q})$ is the set of couplings of $\mathbf{p}$ and $\mathbf{q}$ given by

$$\Pi(\mathbf{p}, \mathbf{q}) = \{\Gamma \in \mathbb{R}_+^{n_x \times n_y} : \Gamma \mathbb{1}_{n_y} = \mathbf{p},\ \Gamma^T \mathbb{1}_{n_x} = \mathbf{q}\}. \tag{3.2}$$

Intuitively, the cost function says how many resources it will take to move point $\mathbf{x}_i$ in the first dataset to point $\mathbf{y}_j$ in the second dataset, and the coupling Γ relates the two discrete measures $\mathbf{p}$ and $\mathbf{q}$ by correspondence probabilities. Each row Γ_i tells us how to split the mass of data point $\mathbf{x}_i$ onto the points $\mathbf{y}_j$ for $j = 1, \ldots, n_y$, and the condition $\Gamma \mathbb{1}_{n_y} = \mathbf{p}$ requires that the sum of each row Γ_i is equal to $\mathbf{p}_i$, the probability of sample $\mathbf{x}_i$. The discrete optimal transport problem finds a coupling matrix, Γ, that

minimizes the cost of moving samples through the linear program:

$$\min_{\Gamma \in \Pi(\mathbf{p},\mathbf{q})} \langle \Gamma, C \rangle. \tag{3.3}$$

Although this problem can be solved with minimum cost flow solvers, it is usually regularized with entropy for more efficient optimization and empirically better results [23]. Entropy diffuses the optimal coupling, meaning that more masses will be split. Thus, the numerical optimal transport problem is

$$\min_{\Gamma \in \Pi(\mathbf{p},\mathbf{q})} \langle \Gamma, C \rangle - \epsilon H(\Gamma), \tag{3.4}$$

where $\epsilon > 0$ and $H(\Gamma)$ is the Shannon entropy ($\sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \Gamma_{ij} \log \Gamma_{ij}$).

Equation 3.4 is a strictly convex optimization problem, and for some unknown vectors $u \in \mathbb{R}^{n_x}$ and $v \in \mathbb{R}^{n_y}$, the solution has the form $\Gamma^* = \text{diag}(u) K \text{diag}(v)$, with $K = \exp\left(-\frac{C}{\epsilon}\right)$, element-wise. This solution can be obtained efficiently via Sinkhorn's algorithm, which iteratively computes

$$u \leftarrow \mathbf{p} \oslash Kv \text{ and } v \leftarrow \mathbf{q} \oslash K^T u, \tag{3.5}$$

where $\oslash$ denotes element-wise division. This derivation immediately follows from solving the corresponding dual problem for Equation 3.4 [70].

3.2.2 Gromov-Wasserstein Optimal Transport

While the classic optimal transport formulation requires us to define a cost function across domains (Equation 3.1), this is difficult to do when working with data from different metric spaces. This is because we cannot directly compare data points with different modalities, such as in the case of multi-omic alignment. Gromov-Wasserstein distance extends optimal transport by comparing distances between data points rather than directly comparing the data points themselves [1] and allows us to work with data from different modalities. Consider the same discrete measures $\mathbf{p}$ and $\mathbf{q}$ as

above, the cost function in the formulation of the optimal transport problem will now be defined over sample-wise pairwise distances $d_x(i,k)$ and $d_y(j,l)$ in the $\mathbf{X}$ and $\mathbf{Y}$ datasets, respectively:

$$GW(\mathbf{p},\mathbf{q}) := \min_{\Gamma\in\Pi(\mathbf{p},\mathbf{q})} \sum_{i,k}^{n_x}\sum_{j,l}^{n_y} L(d_x(i,k),d_y(j,l))\Gamma(i,j)\Gamma(k,l). \tag{3.6}$$

where L indicates the cost function. The main change from basic optimal transport (Equation 3.1) to Gromov-Wasserstein (Equation 3.6) is that we consider the effect of transporting pairs of samples rather than single samples. Intuitively, $L(d_x(i,k), d_y(j,l))$ captures how transporting $\mathbf{x}_i$ to $\mathbf{y}_j$ and $\mathbf{x}_k$ to $\mathbf{y}_l$ would distort the original distances between i and k and between $\mathbf{x}_j$ and $\mathbf{x}_l$. This change ensures that the optimal transport plan Γ will preserve some local geometry.

For solving the Gromov-Wasserstein optimal transport formulation, we compute pairwise distance matrices D^x and D^y for the two domains separately, as well as the fourth order tensor $\mathbf{L}\in\mathbb{R}^{n_x\times n_x\times n_y\times n_y}$, where $\mathbf{L}_{ijkl} = L(D^x_{ik}, D^y_{jl})$. Then, the discrete Gromov-Wasserstein problem can also be expressed as the inner product

$$GW(\mathbf{p},\mathbf{q}) = \min_{\Gamma\in\Pi(\mathbf{p},\mathbf{q})} \langle \mathbf{L}(D^x,D^y)\otimes\Gamma,\Gamma\rangle \tag{3.7}$$

Equation 4.2 is now both non-linear and non-convex and involves operations on a fourth-order tensor, including the $\mathcal{O}(n_x^2 n_y^2)$ operation tensor product $L(D^x,D^y)\otimes\Gamma$ for a naive implementation. Peyré *et al.* show that for some choices of loss function this product can be computed in $\mathcal{O}(n_x^2 n_y + n_x n_y^2)$ cost [69]. In particular, for the case $L = L_2$, the inner product can be computed by

$$\mathbf{L}(D^x,D^y)\otimes\Gamma = (D^x)^2\mathbf{p}\mathbb{1}_{n_y}^T + \mathbb{1}_{n_x}\mathbf{q}^T((D^y)^2)^T - D^x\Gamma(D^y)^T. \tag{3.8}$$

As in the classic optimal transport case, the coupling matrix can be efficiently computed for an entropically regularized optimization problem:

$$GW(\mathbf{p}, \mathbf{q}) = \min_{\Gamma \in \Pi(\mathbf{p}, \mathbf{q})} \langle \mathbf{L}(D^x, D^y) \otimes \Gamma, \Gamma \rangle - \epsilon H(\Gamma). \tag{3.9}$$

Larger values of ϵ lead to an easier optimization problem but also a denser coupling matrix, meaning that solutions will indicate significant correspondences between more data points. Smaller values of ϵ lead to sparser solutions, meaning that the coupling matrix is more likely to find the correct one-to-one correspondences for datasets where there are one-to-one correspondences. However, it also yields a harder (more non-convex) optimization problem [1].

Peyré *et al.* [69] propose using a projected gradient descent approach for optimization, where both the projection and the gradient are taken with respect to Kullback-Leibler divergence. These projections are computed via Sinkhorn iterations. Algorithm 1 in the supplement presents the algorithm for $L = L_2$.

3.2.3 Single-Cell alignment using Optimal Transport (SCOT)

Our method, SCOT, works as follows. First, we compute the pairwise distances on our data in a way similar to [66]. To do this, we use the correlations between data points within each dataset to construct k-NN connectivity graphs. We find that connectivity graphs, which connect nodes with binary edges, empirically work better than weighted edges. This could be because connectivity graphs potentially denoise the data. Next, we compute the shortest path distance on the graph between each pair of nodes via Dijkstra's algorithm. We set the distance of any unconnected nodes to be the maximum finite distance in the graph and normalize the matrix by dividing the elements by this maximum distance. If k is the number of samples, then the k-NN graph is the complete graph, so the corresponding distance matrix is a matrix of all ones. In this case, the distance matrix does not provide information about the local

geometry, so we recommend keeping k small relative to the number of samples to avoid this scenario. We find that our approach is robust to the choice of k (Supplementary Section 1.5)

Since we do not know the true distribution of the original datasets, we follow [1] and empirically set $\mathbf{p}$ and $\mathbf{q}$ to be the uniform distributions on the data points. Then, we solve for the optimal coupling Γ which minimizes Equation 4.4. To implement this method, we use the Python Optimal Transport toolbox (`https://pot.readthedocs.io/en/stable/`) [38].

One of the advantages of using optimal transport is the probabilistic interpretation of the resulting coupling matrix Γ, where the entries of the normalized row $\frac{1}{p_i}\Gamma_i$ are the probabilities that the fixed data point $\mathbf{x}_i$ corresponds to each $\mathbf{y}_j$. However, to use the evaluation metrics previously used in the field and to visualize alignment, we need to project the two datasets into the same space. The Procrustes approach proposed in [1] does not generalize to datasets with different feature and sample dimensions, so we use a barycentric projection:

$$\mathbf{x}_i \mapsto \frac{1}{p_i} \sum_{j=1}^{n_y} \Gamma_{ij} \mathbf{y}_j. \tag{3.10}$$

3.2.4 Alternative Unsupervised Alignment Procedure

In the description of SCOT, the number k for nearest neighbors and the entropy weight ϵ are hyperparameters. One way to set these hyperparameters for optimal alignment is to use some orthogonal correspondence information to select the best alignment either directly [13, 56] or by performing cross-validation [79]. This selection strategy is problematic for truly unsupervised setting, where no correspondence information is available *a priori* upon sequencing separate cell cultures. As a solution, we provide an alternative procedure to learn reasonable alignments based on tracking the Gromov-Wasserstein distance (Equation 4.2). This procedure is based on our

observation that the Gromov-Wasserstein distance serves as a proxy for measuring alignment quality (see Figure 3.5 (A)). In this procedure, we alternate between optimizing ϵ and k to minimize the Gromov-Wasserstein distance between the domains (detailed in Algorithm 2). Although the lowest Gromov-Wasserstein distance is not always the best alignment, it consistently appears to be one of the better alignments.

Algorithm 2: SCOT Alignment with Gromov-Wasserstein OT

Inputs: Datasets $X,\ Y$. Regularization coefficient ϵ. Number of neighbors k.

// Compute graph distances $D_x,\ D_y$; p = Uniform(X), q = Uniform(Y);

$D_{xy} \leftarrow D_x^2 \mathbb{1}_{n_y}^T + \mathbb{1}_{n_x} q (D_x^2)^T$;

while *not converged* **do**

  $\hat{D}_\Gamma \leftarrow D_{xy} - 2 D_x \Gamma D_y^T$;

  // Compute cost matrix

  $u \leftarrow \mathbb{1},\ K \leftarrow \exp\{-\hat{D}_\Gamma / \epsilon\}$; // Perform Sinkhorn iterations

  while *not converged* **do**

    $u \leftarrow p \oslash K v,\ v \leftarrow q^T \oslash K^T u$;

  end

  $\Gamma \leftarrow \mathrm{diag}(u) K \mathrm{diag}(v)$;

end

Return: $n_x \Gamma Y$

Algorithm 3: Unsupervised hyperparameter search procedure

Input: Datasets $X,\ Y$.

$n \leftarrow \min(n_x, n_y),\ k_1 \leftarrow \min(0.2n, 50)$

$\epsilon_1 \leftarrow \text{argmin}\ \ \epsilon \in [10^{-3}, 10^{-2}] \mathrm{SCOT}(X, Y, k_1, \epsilon)$ // Fix k_1 and vary ϵ

// Fix ϵ_1 and vary k

if $n > 250$ **then**

  $k_2 \leftarrow \text{argmin}\ \ k \in [20, 100] \mathrm{SCOT}(X, Y, k, \epsilon_1)$

end

else

  $k_2 \leftarrow \text{argmin}\ \ k \in [0.05n, 0.2n] \mathrm{SCOT}(X, Y, k, \epsilon_1)$

end

// Do a more refined search around k_2 and ϵ_1

$k_{\text{best}}, \epsilon_{\text{best}} \leftarrow$

$\text{argmin}\ \ k \in [k_2 - 5, k_2 + 5],\ \epsilon \in [10^{-0.25}\epsilon_1, 10^{0.25}\epsilon_1] \mathrm{SCOT}(X, Y, k, \epsilon)$

Return: $k_{\text{best}}, \epsilon_{\text{best}}$

3.3 Experimental Setup

3.3.1 Simulated datasets

We follow Liu *et al.* [56] and benchmark SCOT on three different simulations[1]. All three simulations contain two domains with 300 samples that have been non-linearly projected to 1000- and 2000-dimensional feature spaces, respectively. The three simulations are a bifurcation, a Swiss roll, and a circular frustum (Figure 3.3) with points belonging to three different groups. In addition to these three previously existing simulations, we use Splatter [106] to create simulated single-cell RNA sequencing count data, which we call synthetic RNA-seq. We generate 5000 cells with 1000 genes from three cell groups and reduce the count matrix to the five genes with the highest variances. This count matrix is mapped into two new domains with dimensions $p_1 = 50$ and $p_2 = 500$ by multiplying it with two randomly generated matrices, resulting in data with dimensions 5000×50 and 5000×500.

All four datasets were simulated with 1—1 sample-wise correspondences, which are solely used for evaluating model performance. Each domain is projected to a different dimension, so there is no feature-wise correspondence either. In all simulations, we Z-score normalize the features before running the alignment algorithms as in [56].

3.3.2 Single-cell multi-omics datasets

We use two sets of single-cell multi-omics data to demonstrate the applicability of our model to real datasets. Both datasets are generated by co-assays; thus, we have known cell-level correspondence information for benchmarking. The first dataset is generated using the scGEM assay [19], which simultaneously profiles gene expression and DNA methylation. The dataset (Sequence Read Archive accession SRP077853) is derived from human somatic cell samples undergoing conversion to induced pluripotent stem cells (iPSCs) and show a continuous trajectory. This dataset was also used by Cao

[1] `https://noble.gs.washington.edu/proj/mmd-ma/`

et al. [13] to demonstrate the performance of their UnionCom algorithm. We pre-processed the data as described in the original publications [13, 19], and ended up with dimensions are 177×34 for the gene expression data and 177×27 for the chromatin accessibility data.

The second dataset is generated by the SNAREseq assay [18], which links chromatin accessibility with gene expression. The data (Gene Expression Omnibus accession GSE126074) is derived from a mixture of human cell lines: BJ, H1, K562, and GM12878 and show distinct cell type clusters. We pre-process the datasets following Chen *et al.* [18]. The resulting data matrices for the SNARE-seq dataset were of size 1047×19 and 1047×10 for ATAC-seq and RNA-seq, respectively. We unit normalize all real datasets as done in [79].

3.3.3 Evaluation metrics

We compare SCOT with the two state-of-the-art unsupervised single-cell alignment methods MMD-MA [56] and UnionCom [13]. None of these methods use any correspondence information for aligning the datasets. However, all datasets have 1–1 sample-level correspondence information, which we use to quantify the alignment performance through the "fraction of samples closer than the true match" (FOSCTTM) metric introduced by Liu *et al.* [56]. For each domain, we compute the Euclidean distances between a fixed sample point and all the data points in the other domain. Next, we use these distances to compute the fraction of samples that are closer to the fixed sample than its true match. Finally, we average these values for all the samples in both domains. For perfect alignment, all samples would be closest to their true match, yielding an average FOSCTTM of zero. Therefore, a lower average FOSCTTM corresponds to better alignment performance.

Since all the datasets have group-specific (simulations) or cell-type-specific (real experiments) labels, we also adopt the metric used by Cao *et al.* [13] called "label

transfer accuracy" (LTA) to assess the quality of the cell label assignment and to allow for a more direct comparison with their results. This metric measures the ability to correctly transfer sample labels from one domain to another based on their neighborhood in the aligned domain. As described in [13], we train a k-nearest neighbor classifier on one of the domains and predict the sample labels in the other domain. The label transfer accuracy is the proportion of correctly predicted labels, so it ranges from 0 to 1, and higher values indicate good performance. We apply this metric to alignments selected by the FOSCTTM measure.

We benchmark methods under two scenarios: two scenarios: one, where we tune hyperparameters to yield the best alignment results for a given dataset, assuming some correspondence information (as measured by the average FOSCTTM measure) and one, where we assume a fully unsupervised setting, where no correspondence information is available to be used to select hyperparameters. We benchmark methods under two scenarios: when correspondence information exists for validation and when it does not. The first scenario allows us to compare how methods perform with respect to each other in their ideal settings. The second scenario allows us to demonstrate a more realistic use case and shows how methods would perform in a fully unsupervised scenario, where a user would align datasets with no prior correspondence information.

For the first, we choose hyperparameters corresponding to the best alignment as measured by the average FOSCTTM on known correspondences, and we give full details in the next section below. For the second scenario, SCOT offers an alternative automatic hyperparameter tuning procedure as detailed in Section 3.2.4. However, MMD-MA and UnionCom do not provide a similar unsupervised hyperparameter tuning method. Therefore, a user would need to rely on the default hyperparameters of these algorithms. For this scenario, we compare SCOT's alternative tuning procedure with the alignments generated by default hyperparameters for MMD-MA and UnionCom.

3.3.4 Hyperparameter tuning

We run each method over a grid of hyperparameters and select the setting that yields the lowest average FOSCTTM. For SCOT, the grid covers the regularization weight $\epsilon \in \{0.0001, 0.0005, 0.001, 0.005, ..., 0.1\}$ and number of neighbors $k \in \{10, 15, 20, 25, 30, 35, \ldots 100, \frac{1}{6}n_x\}$. We observe empirically that going above $\frac{1}{6}n$ for k does not yield any improvement in alignment.

We pick the hyperparameters for MMD-MA and UnionCom based on the default values and recommended ranges. MMD-MA has three hyperparameters: weights $\lambda_1, \lambda_2 \in \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}\}$ for the terms in the optimization problem and the dimensionality $p \in \{4, 5, 6, 16, 32, 64\}$ of the embedding space. UnionCom requires the user to specify four hyperparameters: the number $kmax \in \{40, 100\}$ of maximum number of neighbors in the graph,the dimensionality $p \in \{4, 5, 6, 16, 32, 64\}$ of the embedding space, the trade-off parameter $\beta \in \{0.1, 1, 10, 15, 20\}$ for the embedding, and a regularization coefficient $\rho \in \{0, 5, 10, 15, 20\}$. We select the embedding dimension $p \in \{16, 32, 64\}$ around the default value of 32 set by UnionCom but also add $p \in \{4, 5, 6\}$ to match the recommended values for MMD-MA. We keep the hyperparameter search space size approximately consistent across the three methods. For each dataset, we present alignment and runtime results for the best performing hyperparameters.

Furthermore, we consider the scenario where correspondence information is unavailable to pick the optimal hyperparameters. For SCOT, we apply the alternative unsupervised alignment algorithm (Algorithm 2 in Supplementary Materials) to align all the datasets. Since MMD-MA and UnionCom do not provide a hyperparameter selection strategy, we rely on the default hyperparameters; we use UnionCom's provided default parameters of $kmax = 40, p = 32, \rho = 10$, and $\beta = 1$, and the center values of MMD-MA's recommended range: $p = 5, \lambda_1 = 10^{-5}$, and $\lambda_2 = 10^{-5}$. We also present the alignment results for all three methods in this fully unsupervised setting.

3.4 Results

We use four simulation datasets and two real-world single-cell sequencing datasets to assess the alignment performance of SCOT. We benchmark it against the two state-of-the-art unsupervised single-cell multi-omics alignment algorithms, MMD-MA and UnionCom, using FOSCTTM and LTA metrics. The former assesses cell-to-cell alignment error and the latter assesses the cell-type grouping accuracy upon alignment.

3.4.1 SCOT successfully aligns the simulated datasets

In this experiment, we align the three simulation datasets from [56], as well as the synthetic single-cell RNA-seq count data generated with Splatter [106]. Prior to alignment, we first select the best performing hyperparameters for each method using the ground-truth correspondence information, as described in Section 3.4.

In Figure 3.3, we visualize the original domains, as well as the alignment performed by SCOT. We color the samples by their domain and cell-type identity. We observe that the global structure is matched, and cells cluster correctly based on cell-type identity. We then sort and plot the FOSCTTM score for each sample in Figure 3.3C. Mean FOSCTTM values are summarized in Table 3.1. We also report the label transfer accuracy values in Table 3.2 when the first domain is used to train a classifier to predict the labels in the second domain. Overall, we observe that SCOT consistently achieves one of the lowest average FOSCTTM scores, thereby demonstrating its ability to recover the correct correspondences. SCOT also consistently yields high label transfer accuracy scores indicating that samples are correctly mapped to their assigned groups.

Table 3.1: Alignment performance by average FOSCTTM measure when the first domain is projected onto the second domain. For real-world datasets, we picked gene expression domain in scGEM and chromatin accessibility domain in SNAREseq to be projected.

	Sim. 1	**Sim. 2**	**Sim. 3**	**Syn. RNA-Seq**	**scGEM**	**SNAREseq**
SCOT	0.085	0.022	**0.009**	**0.001**	**0.192**	**0.150**
MMD-MA	0.124	0.023	0.012	0.112	0.201	**0.150**
UnionCom	**0.083**	**0.016**	0.152	0.038	0.209	0.265

Table 3.2: Alignment performance by label transfer accuracy ($k = 5$) when the first domain (epigenomic domains in real-world datasets) is projected onto the second domain (gene expression domain in real-world datasets).

	Sim. 1	**Sim. 2**	**Sim. 3**	**Syn. RNA-Seq**	**scGEM**	**SNAREseq**
SCOT	0.937	**0.977**	**0.957**	**0.998**	0.576	**0.982**
MMD-MA	0.89	0.783	0.947	0.706	**0.588**	0.942
UnionCom	**0.96**	0.62	0.613	0.997	0.582	0.423

3.4.2 SCOT gives state-of-the-art performance for single-cell multi-omics alignment

Next, we apply our method to real single-cell sequencing data and visualize the alignments in Figure 3.4. To have ground-truth information on cell–cell correspondences solely for benchmarking purposes, we use datasets generated by co-assaying technology. Overall, SCOT gives the lowest average FOSCTTM measure in comparison to MMD-MA and UnionCom (Table 3.1) and recovers accurate 1–1 correspondences in single-cell datasets. For the scGEM data, we report label transfer accuracy using the DNA methylation domain for predicting the cell-type labels in the gene expression domain. For the SNARE-seq dataset, we use the gene expression domain for predicting cell labels in the chromatin accessibility domain (Table 3.2.) SCOT yields the best label transfer accuracy result on SNAREseq dataset and performs comparably to the other methods for scGEM. All methods have higher label transfer accuracy performance on SNAREseq dataset compared to scGEM dataset because SNAREseq dataset contains a mixture of different cell-types that cluster separately, while scGEM dataset contains cells going through a continuous differentiation.

While MMD-MA and UnionCom project both datasets to a shared low-dimensional space, SCOT projects one dataset onto the other. We find that the direction of projection makes no significant difference in performance (Table 3.3).

Table 3.3: Best mean FOSCTTM for each direction of the barycentric projection for all datasets. The method is robust to the direction of the projection.

	Domain 1 onto Domain 2	**Domain 2 onto Domain 1**
Sim. 1	0.085	0.087
Sim. 2	0.022	0.023
Sim. 3	0.009	0.009
Syn. RNA-Seq	0.001	7.68×10^{-5}
scGEM	0.192	0.212
SNARE-seq	0.150	0.151

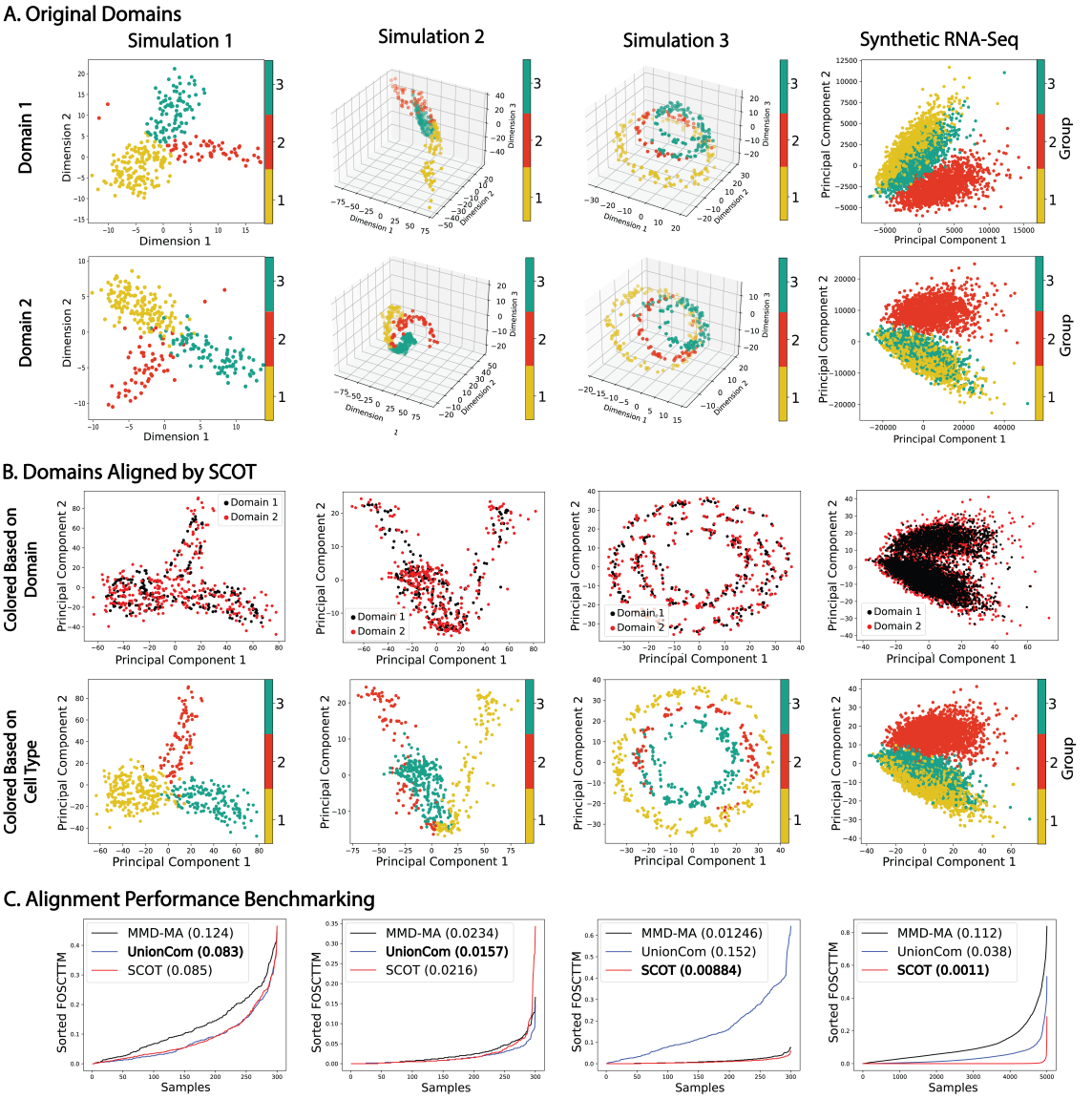

Figure 3.3: Alignment results for simulated datasets.We present the alignment result on four simulations (left to right) - a bifurcation, a Swiss roll, a circular frustum, and synthetic RNA-seq data generated from Splatter [106] **A. Visualization of the dataset before alignment.** Each dataset has two domains to be aligned. **B. Visualization of datasets after alignment by SCOT.** The upper row plots samples colored by domain they come from, while the bottom row shows samples colored by their group (or cell-type) identity. **C. Performance benchmarking**. We plot sorted FOSCTTM measures for alignments performed by SCOT, MMD-MA, and UnionCom for benchmarking. Mean FOSCTTM measures for each alignment and dataset are included in figure legends. Best performing results are bolded.

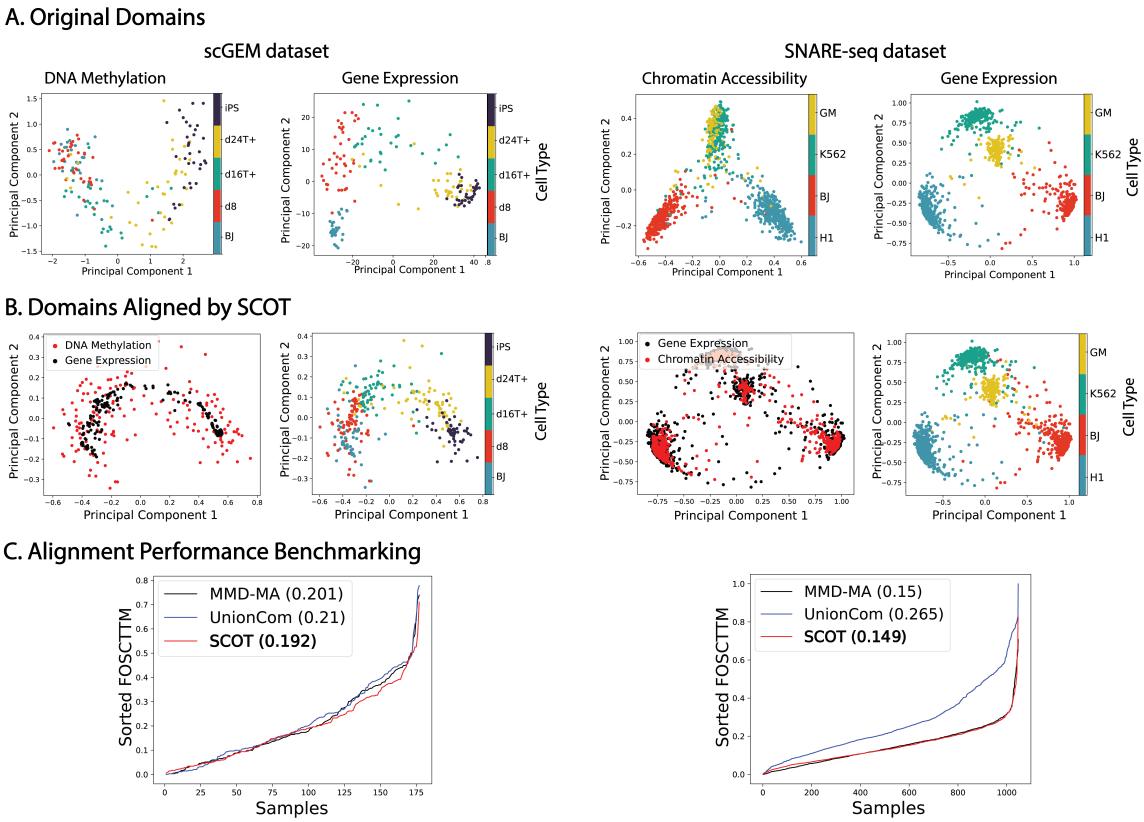

Figure 3.4: Aligning real world single-cell sequencing dataset. **A.** We first visualize the original datasets before alignment. Each dataset has two domains with different sequencing modalities. **Left:** our alignment colored based by domain (plotted in 2D using PCA). **B.** We visualize the aligned datasets after running SCOT. For each dataset, we plot alignments both by coloring data points by domain and by cell-type identity. **C.** We benchmark SCOT against MMD-MA and UnionCom algorithms by comparing FOSCTTM values we get. Graphs here plot sorted FOSCTTM measures and the legend contains average FOSCTTM measures for each alignment.

3.4.3 SCOT's alternative unsupervised hyperparameter tuning procedure achieves quality alignments

We compare the alignment performances in fully unsupervised settings, when we have no validation data on correspondences to use for hyperparameter tuning, as described in Section 3.4. We present the alignment performances, measured by average FOSCTTM measures, in Table 3.4 when using SCOT's alternative self-tuning procedure. In this procedure, hyperparameter choice is guided by the Gromov-Wasserstein

distance, as we have observed a correlation between Gromov-Wasserstein distances between the aligned datasets and alignment quality (Figure 3.5 A). In this unsupervised setting, we use MMD-MA's and UnionCom's default parameters since they lack self-tuning capability. SCOT returns nearly the same alignments for simulated data and only marginally worse alignments for real data. In contrast, MMD-MA and UnionCom show inconsistent alignment performance and fail to align some of the simulated and all real datasets with the default parameter values. Therefore, the proposed procedure could guide a user to an alignment close to the optimal result when no orthogonal information is available.

Table 3.4: Alignment performance by mean FOSCTTM scores in fully unsupervised setting. The hyperparameters for SCOT are chosen by lowest Gromov-Wasserstein distance and the defaul hyperparameters are used for MMD-MA, and UnionCom. Best values are bolded.

	Sim. 1	**Sim. 2**	**Sim. 3**	**Syn. RNA-Seq**	**scGEM**	**SNAREseq**
SCOT (GW)	**0.088**	0.025	**0.009**	**0.001**	**0.209**	**0.218**
MMD-MA	0.125	**0.012**	0.739	0.384	0.437	0.473
UnionCom	0.091	0.028	0.684	0.028	0.691	0.510

Table 3.5: Alignment performance by label transfer accuracy ($k = 5$) in the fully unsupervised setting when the first domain is used for training. The hyperparameters for SCOT are chosen by lowest Gromov-Wasserstein distance and the defaul hyperparameters are used for MMD-MA, and UnionCom. Best values are bolded.

	Sim. 1	**Sim. 2**	**Sim. 3**	**Syn. RNA-Seq**	**scGEM**	**SNAREseq**
SCOT	**0.977**	**0.977**	**0.95**	**0.996**	**0.582**	**0.701**
MMD-MA	0.897	0.957	0.7	0.506	0.237	0.412
UnionCom	0.947	0.947	0.133	0.948	0.107	0.288

3.4.4 SCOT's computation speed scales well with the sample size

We compare SCOT's running times with the baseline methods for the best performing hyperparameters on the synthetic RNA-seq dataset by varying the number of cells to demonstrate how each algorithm scales to larger datasets. While SCOT is

implemented for CPU, both MMD-MA and UnionCom algorithms provide GPU versions, which run faster. Therefore, we use them for benchmarking. We run CPU computations on an Intel Xeon e5-2670 with 16GB memory and GPU computations on a single NVIDIA GTX 1080ti with VRAM of 11GB. SCOT's running time scales similarly to that of MMD-MA, even though SCOT runs on a CPU and MMD-MA runs on a GPU (Figure 3.5 (B)). Both methods scale better than the GPU-based UnionCom implementation.

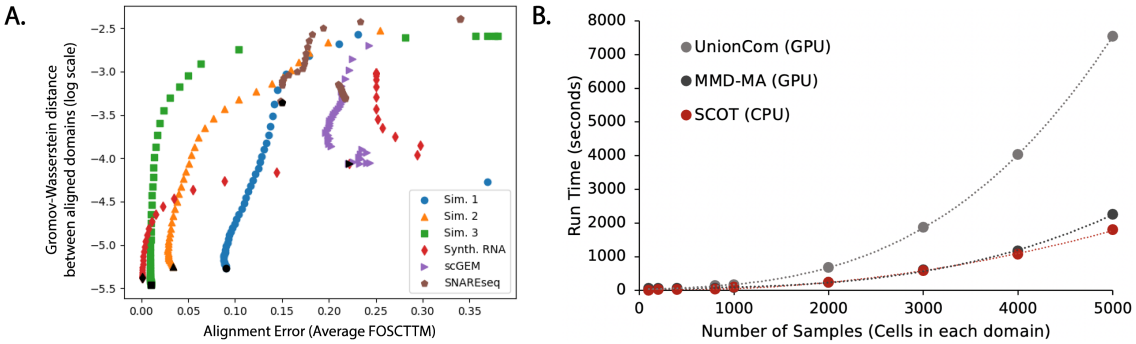

Figure 3.5: A. Runtime comparisons with growing sample size. Dotted lines are polynomial trend lines. B. Relationship between Gromov-Wasserstein distance between the aligned datasets and alignment quality. Lower Gromov-Wasserstein values tend to correspond to better alignments (lower FOSCTTM measures).

3.4.5 Investigating algorithmic choices and hyperparameters of SCOT

To better understand our method, we investigated the effects of different algorithmic choices and hyperparameter combinations on the alignment performance of the real-world datasets. Figure 3.6 shows the range of average FOSCTTM values we receive for alignments with different combinations of k (number of neighbors in $k-$NN graphs and ϵ (entropic regularization coefficient) values for the two real-world sequencing datasets. Overall, we observe that the choice of ϵ tends to make a larger impact on the alignment performance than k. Next, we consider the effect of different algorithmic choices on the alignment performance of SCOT. We compare the final SCOT model with (1) no entropic regularization, (2) using Euclidean distances for intra-domain

distance matrices, and (3) using correlation-based intra-domain distance matrices in lieu of graph distances. For each of these settings, we run alignments for the same combinations of hyperparameters as descrbed in section 3.4 and record the average FOSCTTM measure we receive for each alignment. In Figure 3.7, we compare these in violin plots for scGEM and SNARE-seq datasets. This experiment shows that both entropic regularization and modeling the single-cell datasets as graphs for intra-domain distance computations yield lower FOSCTTM measures, corresponding to higher quality alignments.

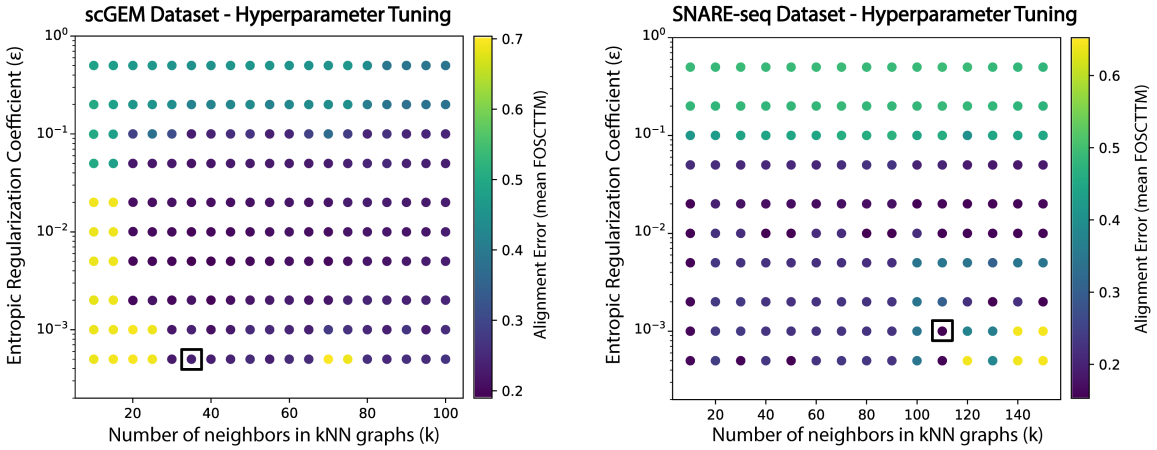

Figure 3.6: Hyperparameter tuning results for scGEM (left) and SNARE-seq (right) datasets. We sweeped a range of values for the two hyperparameters in our model: number of neighbors in $k-$NN graphs, k (on the x-axis), and the entropic regularization coefficient, ϵ (on y-axis). The color of the scattered dots correspond to the average FOSCTTM values we receive for each alignment, with lower values corresponding to better alignments. The hyperparameter combinations that yielded the best FOSCTTM values are in black squares.

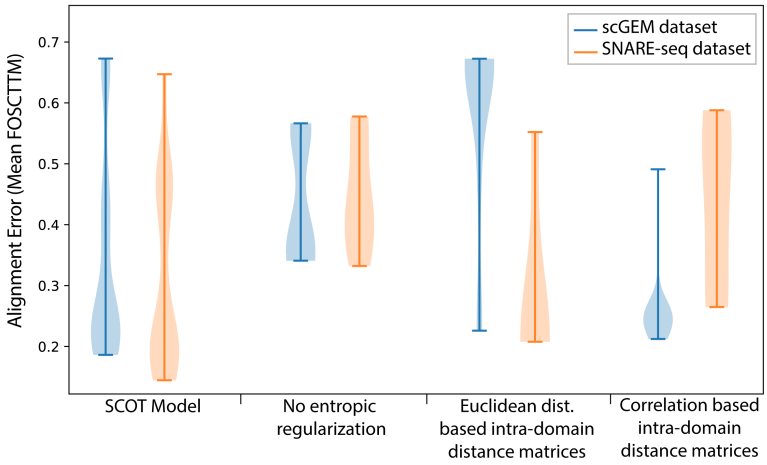

Figure 3.7: Ablation test results. We considered several modifications to algorithmic choices in SCOT and investigated the range of average FOSCTTM values we received in our alignments for scGEM (blue) and SNARE-seq (orange) datasets. The modifications considered are: (1) removing the entropic regularization term from the Gromov-Wasserstein optimal transport objective function, (2) using Euclidean distances for intra-domain distance and (3) using correlation-based distances instead of graph distances for the intra-domain distance matrices.

3.5 Discussion

We have demonstrated that SCOT, which uses Gromov Wasserstein optimal transport for unsupervised single-cell multi-omics data integration, performs on par with UnionCom and MMD-MA when sample correspondence information is available for hyperparameter tuning and shows advantages in other scenarios and aspects. Our formulation of a coupling matrix based on matching graph distances is somewhat similar to UnionCom's initial step; however, UnionCom only matches sample-to-sample distances, while Gromov-Wasserstein distance considers the cost of moving pairs of points, enabling our method to better preserve local geometry. Additionally, SCOT performs global alignment of the marginal distributions, which is similar to how MMD-MA uses the MMD term to ensure that the two distributions agree globally in the latent space. We hypothesize that these properties result in SCOT's state-of-the-art performance. Furthermore, SCOT's optimization runs in less time

and with fewer hyperparameters, and the Gromov-Wasserstein distance can guide the user to choose an alignment when no validation information exists. Therefore, unlike other methods, SCOT easily yields high quality alignments in the realistic fully unsupervised setting.

While barycentric projection provides a way to visualize the alignment, it assumes that cells in one dataset should be mapped to the convex hull of the other dataset. In the next chapter, we reformulate SCOT with unbalanced Gromov-Wasserstein optimal transport, which takes care of outliers as well as under- or over-represented groups. There are also other ways to use the coupling matrix to infer alignment such as using it with other dimension reduction methods like t-SNE (as in UnionCom) to align the manifolds while embedding them both into a new space. Alternatively, depending on the application, a projection may not be required; it may be sufficient to have probabilities relating the samples to one another. Future work could develop effective ways to utilize the coupling matrix and extend our framework to handle more than two alignments at a time.

Chapter 4

SCOTv2: Unbalanced Multi-domain Single-cell Alignment

4.1 Introduction

In Chapter 3, we discussed unsupervised integration of separately profiled (i.e. unpaired) single-cell multi-omics datasets, and introduced an algorithm to address this challenge, **S**ingle-**c**ell alignment with **O**ptimal **T**ransport (SCOT). The unsupervised methods [13, 35, 56, 83] discussed in Chapter 3, including our work SCOT [25, 28], have shown good performance for integrating different single-cell measurement domains when tested on datasets obtained from co-assays. Since these methods were only evaluated on co-assays (with 1–1 correspondence between cells across domains), our understanding of their performance on datasets obtained from experiments that are not co-assays is limited. Such experiments perform separate sampling to measure distinct genomic features, like gene expression and 3D chromatin conformation. As a result of this sampling, their datasets can consist of varying proportions of cell-types across different measurements, creating cell-type imbalance and lacking 1–1 cell correspondences. We hypothesize that alignment methods that perform well on

co-assay datasets may not effectively handle the differences in cell-type proportions of the commonly available non-co-assay datasets. Indeed, a recent method, Pamona [14], extended our SCOT framework and used partial Gromov-Wasserstein (GW) optimal transport to allow for missing or underrepresented cell-types in one domain when performing alignment. The paper showed that current integration methods [13, 25, 28, 56, 83] tend to perform worse under such settings.

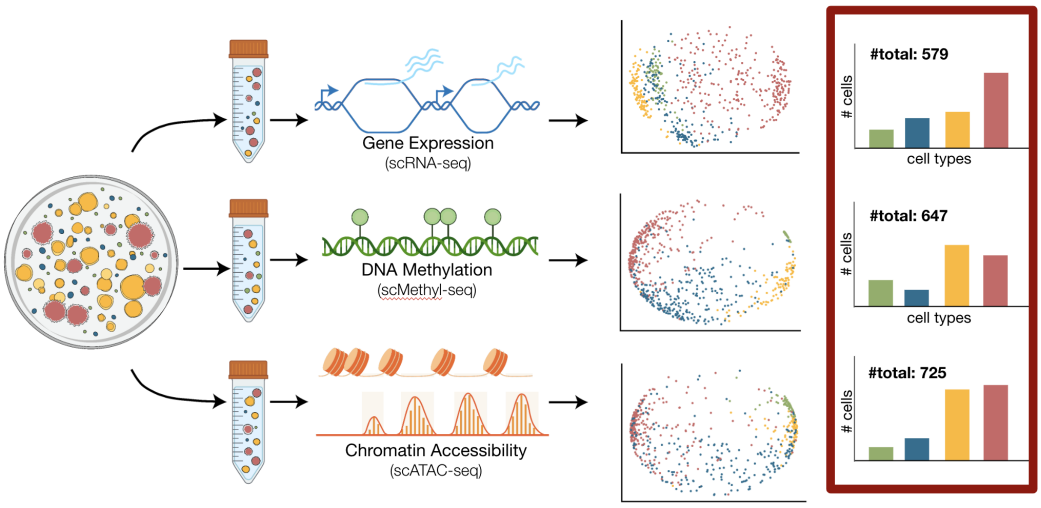

Figure 4.1: An example of the cell-type representation imbalance observed in real-world unpaired single-cell multi-omic datasets. This particular example is from scNMT-seq dataset. While this dataset is generated via a co-assaying technology (i.e. paired multi-omic dataset), the cell-type representation disproportion arises because different number of cells are retained after the quality control procedure is carried out for each measurement modality.

4.2 Our contributions

We present SCOTv2, a novel extension of SCOT that can effectively align both co-assay and non-co-assay datasets using a single framework. It uses *unbalanced* GW optimal transport to align datasets with disproportionate cell-types while only introducing one additional hyperparameter. This unbalanced framework relaxes the constraint that each point must be mapped with its original mass during the optimal

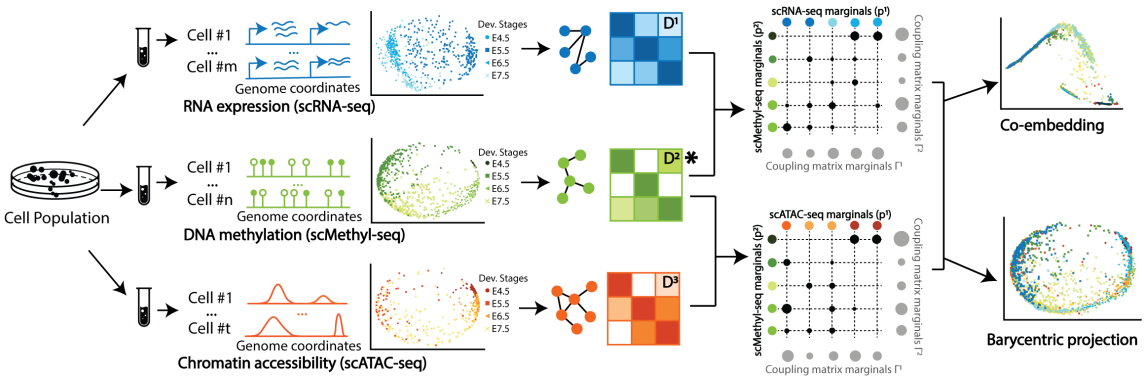

Figure 4.2: Overview of SCOTv2 on scNMT-seq dataset [20], which contains unbalanced cell-type representation across three domains - RNA expression, chromatin accessibility, and DNA methylation. SCOTv2 selects an anchor domain (denoted with *) and aligns other measurements to it. First, it computes intra-domain distances matrices D^m for $m = 1, 2, 3$, which are used to solve for correspondence matrices between the anchor and other domains. The circle sizes in the matrices depict the magnitude of the correspondence probabilities or how much mass to transport. Unbalanced GW relaxes the mass conservation constraint, so the transport map does not need to move each point with its original mass. Finally, it either co-embeds the domains into a common space or uses barycentric projections to project them onto the anchor domain.

transport. Specifically, an underrepresented cell-type in one domain can be transported with more mass to match the proportion of that cell-type in the other domain and vice-versa. The SCOTv2 framework is summarized in Figure 4.2. We demonstrate that SCOTv2 aligns datasets with imbalance in cell-type representations better than state-of-the-art baselines and computationally scales as well as the fastest methods. Furthermore, we extend SCOTv2 to integrate single-cell datasets with more than two measurements, making it a multi-omics alignment tool. We perform alignments of five real-world single-cell datasets, with both simulated and natural cell-type imbalance as well as two and more than two domains ($M \geq 2$), demonstrating SCOTv2's applicability across a wide range of scenarios. Finally, similar to the previous version, we present a self-tuning heuristic process to select hyperparameters for SCOTv2 without any corresponding information like cell-type annotations or matching cells or features in truly unsupervised settings.

4.3 Method

Optimal transport finds the most cost-effective way to move data points from one domain to another. One can imagine it as the problem of moving a pile of sand to fill in a hole through the least amount of work. Our previous framework SCOT [25, 28] uses Gromov-Wasserstein optimal transport, which preserves local geometry when moving data points from one domain to another. The output of SCOT is a matrix of probabilities that represent how likely it is that data points from one modality correspond to data points in the other.

Here, we reintroduce the SCOT formulation to integrate M domains (or single-cell measurements) $X^m = (x_1^m, x_2^m, \ldots x_{n_m}^m) \in \mathbb{R}^{d_m}$ for $m = 1, \ldots M$ with n_m data points (or cells) each. For each dataset, we define a marginal distribution p^m, which can be written as an empirical distribution over the data points:

$$p^m = \sum_{i=1}^{n_m} p_i^m \delta_{x_i}. \tag{4.1}$$

Here, δ_{x_i} is the Dirac measure. For SCOT, we choose these distributions to be uniform over the data.

Gromov-Wasserstein optimal transport performs the transport operation by comparing distances between samples rather than directly comparing the samples themselves [1]. Therefore, for each dataset, we compute the intra-domain distance matrix D^m. Next, we construct k-NN graphs based on correlations between data points and use Dijkstra's algorithm to compute the shortest path distance on the graph between each pair of nodes. Finally, we connect all unconnected nodes by the maximum finite distance in the graph and set D^m to be the matrix resulting from normalizing the distances by this maximum.

For two datasets and a given cost function $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, we compute the fourth-order tensor $\mathbf{L} \in \mathbb{R}^{n_x \times n_x \times n_y \times n_y}$, where $\mathbf{L}_{ijkl} = L(D^1_{ik}, D^2_{jl})$. Intuitively, L quantifies

how transporting a pair of points x_i^1, x_k^1 onto another pair across domains, x_j^2, x_l^2, distorts the original intra-domain distances and helps to preserve local geometry. Then, the discrete Gromov-Wasserstein problem between p^1 and p^2 is,

$$GW(p^1, p^2) = \min_{\Gamma \in \Pi(p^1, p^2)} \sum_{i,j,k,l} \mathbf{L}_{ijkl} \Gamma_{ij} \Gamma_{kl}, \tag{4.2}$$

where Γ is a coupling matrix from the set:

$$\Pi(p^1, p^2) = \{\Gamma \in \mathbb{R}_+^{n_1 \times n_2} : \Gamma \mathbb{1}_{n_2} = p_1,\ \Gamma^T \mathbb{1}_{n_1} = p_2\}. \tag{4.3}$$

One of the advantages of using optimal transport is the probabilistic interpretation of the resulting coupling matrix Γ, where the entries of the normalized row $\frac{1}{p_i}\Gamma_i$ are the probabilities that the fixed data point x_i corresponds to each y_j. Each entry Γ_{ij} describes how much of the mass of x_i should be mapped to y_j.

To make this problem more computationally tractable, we solve the entropically regularized version:

$$GW_\epsilon(p^1, p^2) = \min_{\Gamma \in \Pi(p^1, p^2)} \langle \mathbf{L}(D^1, D^2) \otimes \Gamma, \Gamma \rangle - \epsilon H(\Gamma). \tag{4.4}$$

where $\epsilon > 0$ and $H(\Gamma)$ is the Shannon entropy defined as $H(\Gamma) = \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} \Gamma_{ij} \log \Gamma_{ij}$. Larger values of ϵ make the problem more convex but also lead to a denser coupling matrix, meaning there are more correspondences between samples. In SCOT, we use the cost function $L = L_2$.

4.3.1 Unbalanced Optimal Transport of SCOTv2

Our proposed solution to align datasets with different numbers of samples or proportions of cell-types is to use unbalanced optimal transport, which adds divergence terms to allow for mass variations in the marginals [55, 86]. We follow Séjourné *et al*

[86], and use the Kullback-Leibler divergence ,

$$\mathrm{KL}(p||q) = \sum_x p(x) \log\left(\frac{p(x)}{q(x)}\right), \tag{4.5}$$

to measure the difference between the marginals of the coupling Γ and the input marginals p^1 and p^2. Thus, we solve the unbalanced GW problem:

$$GW_{\epsilon,\rho}(p^1, p^2) = \min_{\Gamma \geq 0} \langle \mathbf{L}(D^1, D^2) \otimes \Gamma, \Gamma \rangle - \epsilon H(\Gamma) + \rho \mathrm{KL}(\Gamma \mathbb{1}_{n_2} || p^1) + \rho \mathrm{KL}(\Gamma^T \mathbb{1}_{n_1} || p^2), \tag{4.6}$$

where $\rho > 0$ is a hyperparameter that controls the marginal relaxation. When ρ is large, the marginals of Γ should be close to p^1 and p^2, and when ρ is small, the marginals of Γ may differ more, allowing each point to transport with more or less mass than it originally had. We demonstrate the effects of this relaxation term in Figure 4.3. See Supplementary Algorithm 4 for details.

4.3.2 Extending SCOTv2 for Multi-Domain Alignment

To align more than two datasets ($M > 2$), we use one domain as an anchor to align the other domains. The anchor should be the domain with the clearest biological structures, for example, a dataset with the best-defined cell-type clusters. We propose selecting the anchor via the kNN graph used to compute D^m. For every node x_i^m in the graph, we calculate the average of the k neighboring node values $\mathcal{N}_k(x_i^m)$. Next, we measure the difference between this average and the true value of the node. This difference reflects how well the averaged neighborhood represents the given node. We then average these differences across the graph and select the domain with the lowest averaged difference as the anchor. Intuitively, we select the anchor whose kNN graph best reflects its dataset. Suppose X^1 is the anchor dataset. Then, for $m = 2, 3, \ldots, N$, we compute the coupling matrix Γ^m according to Equation 4.4.

To have all of the datasets aligned in the same domain, we can either use barycentric projection to project each X^m for $m = 2, 3, \ldots, M$ onto X^1 or find a shared embedding space as described in Section 4.3.3. In the first iteration of SCOT, we used a barycentric projection to align and project one dataset onto the other. Due to the marginal relaxation, we now search for a non-negative $n_1 \times n_m$ dimensional matrix Γ instead of $\Gamma \in \Pi(p^1, p^m)$. Because of this change, the adjusted barycentric projection is:

$$x_i^m \mapsto \frac{\sum_{j=1}^{n_1} \Gamma_{ij}^m x_j^1}{\sum_{j=1}^{n_1} \Gamma_{ij}^m}. \tag{4.7}$$

4.3.3 Embedding with the Coupling Matrix

Other methods such as MMD-MA and UnionCom align datasets by embedding them into a common latent space of dimension $p \leq \min_{m=1,\ldots,M} d_m$. Here d_m represents the original dimension size of measurement (or domain) m. Embedding the datasets in a new space often leads to a better alignment as it introduces the additional benefits of dimension reduction, allowing more meaningful structures in the datasets such as cell-types to be more prevalent. Due to these benefits, we also enable the embedding option through a modification of the t-SNE method proposed by UnionCom [13]. For each domain m, we compute P^m, an $n_m \times n_m$ cell-to-cell transition matrix; each entry $P_{j|i}^m$ is the conditional probability that a data point x_i^m would pick x_j^m as its neighbor when chosen according a Gaussian distribution centered at x_i^m. Similarly, for the lower-dimensional embeddings, we compute a cell-to-cell probability matrix $Q^{m'}$ through a Student-t distribution. The full descriptions of P^m and $Q^{m'}$ are given in Supplementary Section 4.3.4.

Then, to jointly embed all domains through the anchor domain X^1, the optimization problem is:

$$\min_{X^{1'},\ldots,X^{M'}} \sum_{m=1}^{M} \mathrm{KL}(P^m || Q^{m'}) + \beta \sum_{m=2}^{M} ||X^{1'} - X^{m'}(\Gamma^m)^T||_F^2, \tag{4.8}$$

where $X^{m'}$ is the lower dimensional embedding of X^m, and Γ^m is the coupling matrix from solving Equation 4.6 for $m = 2, \ldots, M$. These two terms seek to find an embedding that both preserves the local geometry in the original domain and aligns the domains according to the correspondence found by GW. The intuition behind the term $\mathrm{KL}(P^m||Q^{m'})$ is very similar to that of GW; if two points have a high transition probability in the original space, then they should also have a high transition probability in the latent space. The term $||X^{1'} - X^{m'}(\Gamma^m)^T||_F^2$ measures how well aligned the new embeddings $X^{1'}$ and $X^{m'}$ are according to the prescribed coupling matrix Γ^m. Finally, $\beta > 0$ controls the trade-off between preserving the original geometry with the KL term and enforcing the alignment found with GW. We solve this optimization problem using gradient descent from UnionCom with a default latent space dimension size $p = 3$ [13]. The overall SCOTv2 method is presented as Algorithm 5.

4.3.4 Embedding Method Details

The full details of t-SNE can be found in [90]. For each domain m, we compute P^m, an $n_m \times n_m$ cell-to-cell transition matrix; each entry $P^m_{j|i}$ is the conditional probability that a data point x_i^m would pick x_j^m as its neighbor when chosen according a Gaussian distribution centered at x_i^m:

$$P^m_{j|i} = \frac{\exp(-||x_i^m - x_j^m||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||x_i^m - x_k^m||^2/2\sigma_i^2)}. \tag{4.9}$$

The bandwidth σ_i is chosen according to the density of the data points through a binary search for the value of σ_i that achieves the user-supplied perplexity value. P^m is computed by averaging $P^m_{i|j}$ and $P^m_{j|i}$ to give more weight to outlier points:

$$P^m_{ij} = \frac{P^m_{i|j} + P^m_{j|i}}{2n_m} \tag{4.10}$$

Algorithm 4: Pseudocode for Unbalanced GW Optimal Transport (UGWOT)

Input: Marginal probabilities p^1 and p^2, intra-domain distance matrices D^1 and D^2, relaxation coefficient ρ, regularization coefficient ϵ
Initialize the coupling matrix: $\Gamma = \pi = p^1 \otimes p^2$
while Γ *not converged* **do**
  $\Gamma \leftarrow \pi$
  $\Gamma_{(mass)} \leftarrow \sum_{i,j} \Gamma_{i,j}$ $\tilde{\epsilon} \leftarrow \Gamma_{(mass)}\epsilon$, $\tilde{\rho} \leftarrow \Gamma_{(mass)}\rho$
  // Compute cost C:
  $\Gamma^1 \leftarrow \Gamma \mathbb{1}_{n_2}$, $\Gamma^2 \leftarrow \Gamma^T \mathbb{1}_{n_1}$
  $A \leftarrow (D^1)^{\circ 2}\Gamma^1$, $B \leftarrow (D^2)^{\circ 2}\Gamma^2$
  $D \leftarrow D^1 \Gamma D^2$
  $E \leftarrow \epsilon \sum_{ij} \log\left(\frac{\Gamma_{i,j}}{p_i^1 p_j^2}\right)\Gamma_{i,j} + \rho\left(\sum_i \log\left(\frac{\Gamma_i^1}{p_i^1}\right)\Gamma_i^1 + \sum_j \log\left(\frac{\Gamma_j^2}{p_j^2}\right)\Gamma_j^2\right)$
  $C \leftarrow A + B - 2D + E$
  // Perform Sinkhorn iterations
  while (u, v) *not converged* **do**
    $u \leftarrow -\frac{\epsilon\tilde{\rho}}{\tilde{\epsilon}+\tilde{\rho}} \log\left[\sum_{i,j} \exp(v_j - C_{ij})/\tilde{\epsilon} + \log p^2\right]$
    $v \leftarrow -\frac{\epsilon\tilde{\rho}}{\tilde{\epsilon}+\tilde{\rho}} \log\left[\sum_{i,j} \exp(u_i - C_{ij})/\tilde{\epsilon} + \log p^1\right]$
  end
  // Update: $\pi_{ij} \leftarrow \exp\left[u_i + v_j - C_{ij}\right] p_i^1 p_j^2$
  // Rescale: $\pi \leftarrow \sqrt{\Gamma_{(mass)}/\pi_{(mass)}}\pi$
end
Return: Γ

Then, to jointly embed all domains through the anchor domain X^1, the optimization problem is:

$$\min_{X^{1'},\ldots,X^{M'}} \sum_{m=1}^{M} \mathrm{KL}(P^m||Q^{m'}) + \beta \sum_{m=2}^{M} ||X^{1'} - X^{m'}(\Gamma^m)^T||_F^2, \tag{4.11}$$

where $X^{m'}$ is the lower dimensional embedding of X^m, P^m is defined as in Equation 4.9, and Γ^m is the coupling matrix from solving Equation 4.6 for $m = 1, 2, \ldots, M$, $X^{m'}$. The probability matrix Q^m is computed through a Student-t distribution with one degree of freedom:

$$Q_{ij}^{m'} = \frac{(1 + ||x_i^{m'} - x_j^{m'}||)^{-1}}{\sum_{k \neq l} 1 + (||x_k^{m'} - x_l^{m'}||)^{-1}}. \tag{4.12}$$

Algorithm 5: Pseudocode for SCOTv2 Algorithm

Input: Datasets $X^1, \ldots, X^M$, number of neighbors in nearest neighbor graphs k, entropic regularization coefficient ϵ, mass conservation relaxation coefficient ρ.
for $m = 1, \ldots, M$ **do**
  // Initialize marginal probabilities: $p^m \leftarrow \text{Uniform}(X^m)$;
  //Construct G^m, a $k-$NN graph based on pairwise correlations
  // Compute intra-domain distance matrix D^m on G^m with Dijsktra's algorithm.
  // Compute a "neighborbood correlation" score, c^m:
  $c^m = \frac{1}{n_m} \sum_{i=1}^{n_m} \frac{1}{k} \sum_{x_j^m \in \mathcal{N}_k(x_i^m)} \text{corr}(x_j^m, x_i^m)$
end
// Select an anchor domain X^{m*}: $m^* =_{m=1,\ldots M} c^m$
for $m = 1, \ldots, M$ $(m \neq m^*)$ **do**
  // Compute pairwise coupling matrices between the anchor domain X^{m*} and all other domains:
  $\Gamma^m \leftarrow GW_{\epsilon,\rho}(p^m, p^{m*})$
  if *Barycentric projection* **then**
    $x_i^{m'} \leftarrow \frac{\sum_{j=1}^{n_1} \Gamma_{ij}^m x_j^{m*}}{\sum_{j=1}^{n_1} \Gamma_{ij}^m}$
  end
  else
    // Find shared embedding (e.g. via modified t-SNE as detailed below)
    $X^{1'} \ldots X^{M'} \leftarrow$
    $\min_{X^{m'}, \ldots, X^{M'}} \sum_{m=1}^{M} \text{KL}(P^m || Q^{m'}) + \beta \sum_{m \neq m*} ||X^{m*'} - X^{m'}(\Gamma^m)^T||_F^2$
  end
end
Return: Aligned datasets, $X^{1'} \ldots X^{M'}$.

The intuition behind the cost $\text{KL}(P^m || Q^{m'})$ is very similar to that of GW; if two points have a high transition probability in the original space, then they should also have a high transition probability in the latent space.

Additionally to the t-SNE co-embedding, we give users a choice to co-embed datasets in a shared $d-$dimensional space in a similar fashion to Pamona [14], where, for the anchor dataset $\mathbf{X}$ and non-anchor datasets $\mathbf{Y}^i$ for $i = 1, ..., m$ datasets. For this, we compute the graph Laplacian matrices $\mathbf{L_X}$ and $\mathbf{L}_\mathbf{Y}^i$ and introduce rotation

invariant constraints to find embeddings $\mathbf{X^e}, \mathbf{Y^e}$ that yields:

$$\max_{\mathbf{X^e},\mathbf{Y^e}} \text{trace}(\mathbf{X^e \Gamma^e Y^{e^T}}) \tag{4.13}$$

$$\text{such that } \mathbf{X}^e \mathbf{S_{xx}} \mathbf{X^{e^T}} = \mathbf{I}, \mathbf{Y^e S_{yy} Y^{e^T}} = \mathbf{I} \tag{4.14}$$

where

$$\mathbf{Y}^e = [\mathbf{Y}^{1e}, ..., \mathbf{Y}^{me}], \mathbf{S_{xx}} = \sum_{\mathbf{i=1}}^{\mathbf{m}} (\mathbf{L_x} + \lambda \mathbf{\Sigma_x^i}), \tag{4.15}$$

$$\mathbf{\Sigma_X^i} = \text{diag}(\mathbf{\Gamma}^i \mathbb{1}_{n_i}), \mathbf{\Sigma_Y^i} = \text{diag}(\mathbb{1}_{n_x}^T \mathbf{\Gamma}^i), i = 1, ..., n_y \tag{4.16}$$

$$\mathbf{\Gamma}^e = [\mathbf{\Gamma}^1, ..., \mathbf{\Gamma}^m]^T \tag{4.17}$$

$$\mathbf{S}_{yy} = \text{diag}(\mathbf{L}_y^1 + \lambda \mathbf{\Sigma_Y^1}, ..., \mathbf{L}_y^1 + \lambda \mathbf{\Sigma_Y^1}) \tag{4.18}$$

with λ being a hyperparameter. Equation 4.13 is optimized using the eigenvalue decomposition method as in Pamona [14]. Implementation can be found in Section 7.

4.3.5 Heuristic process for self-tuning hyperparameters

SCOTv2 has three hyperparameters: (1) k for the number of neighbors to consider in nearest neighbor graphs, (2) the weight of the entropic regularization term, ϵ, and (3) the coefficient of the mass relaxation constraint, ρ. The barycentric projection of one domain onto another does not require any hyperparameters. However, jointly embedding the domains in a latent space requires selecting the dimension p.

Ideally, orthogonal correspondence information such as 1–1 correspondences and cell-type labels can guide hyperparameter tuning as validation. However, such information is hard to obtain in most cases. First, no validation data on cell-to-cell correspondences exists for non-co-assay datasets. Second, it is challenging to infer cell-types for certain sequencing domains such as 3D chromatin conformation. Lastly, the cell-type annotations may not always agree across single-cell domains.

We provide a heuristic to self-tune hyperparameters in the completely unsupervised setting. We first choose a k for the neighborhood graphs that yields a high average correlation value between the neighborhood predicted values and measured genomic values of the graph nodes. This step is the same as the one used to select the anchor domain for multi-omics alignment in Section 4.3.2. Next, we choose ϵ and ρ values that minimize the Gromov-Wasserstein distance between the aligned datasets. Algorithm 6 gives the details of this procedure.

Algorithm 6: Unsupervised hyperparameter search procedure

Input: Datasets $X^1, \ldots, X^M$.
// Find k for each domain
for $m = 1, \ldots, M$ **do**
  $k^m = \underset{k \in \{10,20,\ldots,150\}}{\operatorname{argmax}} \frac{1}{n_m} \sum_{i=1}^{n_m} \frac{1}{k} \sum_{x_j^m \in \mathcal{N}_k(x_i^m)} \operatorname{corr}(x_j^m, x_i^m)$
  // Use k^m to compute D^m
end
// Use the GW distance to pick ρ and ϵ
for $m = 2, \ldots, M$ **do**
  $\epsilon^m, \rho^m = \operatorname{argmin}\ \epsilon, \rho GW_{\epsilon,\rho}(\mathbb{1}_{n_1}, \mathbb{1}_{n_m})$
end
Return: k^m, ϵ^m, ρ^m.

4.4 Experimental Setup

4.4.1 Datasets

We evaluate SCOTv2 on single-cell datasets with disproportionate cell-types using two schemes. (1) We subsample different cell-types in co-assay datasets to simulate cell-type representation disparities between sequencing modalities. (2) We select real-world separately sequenced single-cell multi-omics datasets, which lack 1–1 cell correspondences and have different cell-type proportions across modalities due to the sampling procedure. Additionally, we present results on the original co-assay datasets

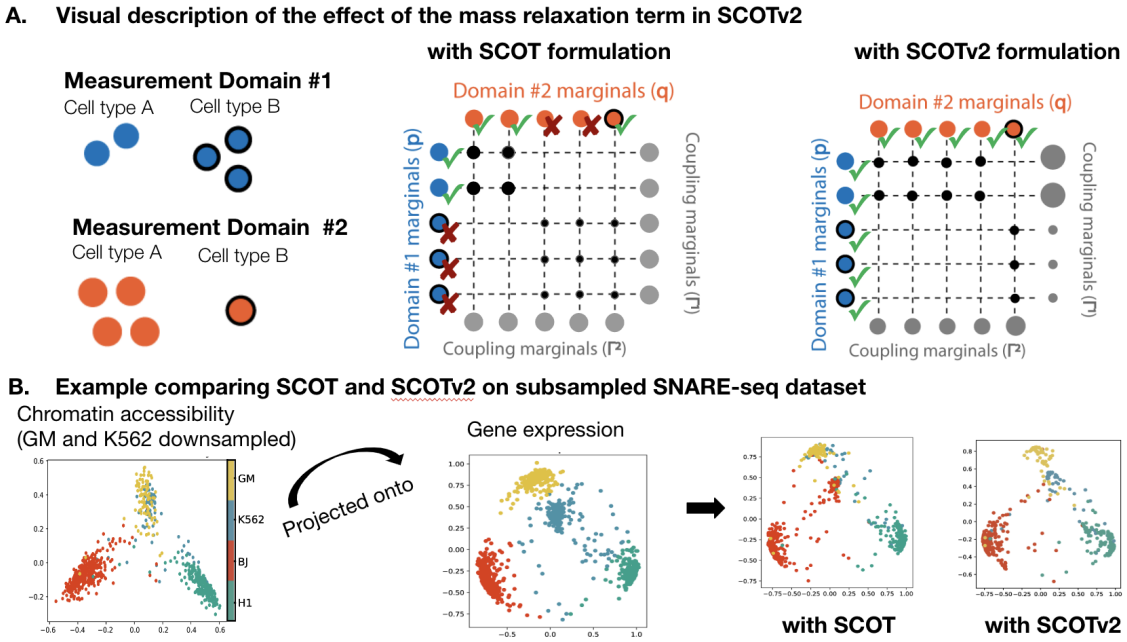

Figure 4.3: Schematic visualizing the effect of the mass relaxation term in SCOTv2 (unbalanced Gromov-Wasserstein optimal transport). A. By allowing for the marginal distributions of the coupling matrix to diverge from the dataset marginals, we let the mass of each datapoint to be locally modified during transportation to better match cells from similar cell types, yielding better alignments for datasets with disproportionate cell-type representation. B. An example comparing SCOT and SCOTv2 alignments on SNARE-seq dataset alignment, with subsampled cell-type clusters in the chromatin accessibility domain to simulate cell-type imbalance. Notice that SCOT moves cells from over-represented cell-types (e.g. BJ) in the place of underrepresented cell-types (e.g. K562), while SCOTv2 more correctly aligns cells.

with 1–1 cell correspondence to demonstrate the flexibility of SCOTv2 across balanced and unbalanced single-cell datasets.

Co-assay single-cell datasets with 1–1 cell correspondence

We use three co-assay datasets to validate our model, sequenced by SNARE-seq, scGEM, and scNMT technologies. SNARE-seq is a two-modality sequencing technology that simultaneously captures the chromatin accessibility and transcriptional profiles of cells [18]. This dataset contains a total of 1047 cells from four cell lines: BJ (human fibroblast cells), H1 (human embryonic cells), K562 (human erythroleukemia

cells), and GM12878 (human lymphoblastoid cells) (Gene Expression Omnibus access code: GSE126074). We follow the same data preprocessing steps outlined by Chen *et al.* [18]. The scGEM technology is a three-modality sequencing technology that profiles the genetic sequence, gene expression, and DNA methylation states in the same cell [19]. The dataset we use is derived from human somatic cell samples undergoing conversion to induced pluripotent stem cells (Sequence Read Archive accession code SRP077853) [19]. We access the preprocessed data provided by Welch *et al.* [97], which only contains the gene expression and DNA methylation modalities [1]. The dataset sequenced by scNMT-seq method [5] contains three modalities of genomic data: gene expression, DNA methylation, and chromatin accessibility, from mouse gastrulation samples, going through the Carnegie stages of vertebrate development (Gene Expression Omnibus access code: GSE109262). We access the preprocessed data through the scripts[2] provided by the authors. While the SNARE-seq and scGEM datasets contain the same number of cells across measurements, scNMT-seq modalities contain different cell-type proportions after preprocessing due to varying noise levels in measurements. Supplementary Table 4.1 lists the number of cells belonging to different cell-types in each domain for scNMT-seq dataset.

Single-cell datasets with simulated cell-type imbalance.

To test alignment performance sensitivity to different levels and types of cell-type proportion disparities across modalities, we generate simulation datasets by subsampling SNARE-seq and scGEM co-sequencing datasets in two ways. (1) We remove a cell-type from one modality. (2) We reduce the proportion of a cell-type in one modality by subsampling it at 50% and another cell-type in the other modality by subsampling it at 75%. We simulate this setting to test how the alignment methods will behave when multiple cell-types have disproportionate representation at different

[1]Preprocessed data for the scGEM dataset accessed here: `https://github.com/jw156605/MATCHER`
[2]Preprocessing scripts for the scNMT-seq data accessed here: `https://github.com/PMBio/scNMT-seq/`

levels (for example, half or quarter percentage of cell-types missing) across modalities.

For these cases, we uniformly pick at random which cell-type to subsample or remove. Specifically, for scGEM in simulation case (1), we remove "d16T+" cells in the DNA methylation domain while retaining the original gene expression domain, and remove the "d24T+" cells in the gene expression domain while retaining the original DNA methylation domain. For the SNARE-seq dataset, we remove "GM" cells in the gene expression domain and "K562" in the chromatin accessibility domain. In simulation case (2), we subsample the "d8" cluster of the scGEM dataset at 75% in the gene expression modality and the "d16T+" cluster at 50% in the DNA methylation modality. For SNARE-seq, we subsample the "H1" cluster at 75% and the "K562" cluster at 50% in the gene expression and chromatin accessibility domains, respectively.

Single-cell datasets without 1—1 correspondences

We also align non-co-assay datasets, containing separately sequenced single-cell -omic measurements. Bonora *et al.* generated the first dataset we use, "sciOmics" [9]. This dataset consists of sciRNA-seq, sciATAC-seq, and sciHiC measurements, capturing gene expression, chromatin accessibility, and 3D chromosomal conformation profiles of mouse embryonic stem cells undergoing differentiation. The measurements were taken at five stages: days 0, 3, 7, 11, and as fully differentiated neural progenitor cells (NPCs). The second non-co-assay dataset, "MEC," contains gene expression and chromatin accessibility measurements taken using the 10X Chromium scRNA-seq and scATAC-seq technologies on mouse mammary epithelial cells (MEC). Since each modality consists of separately sampled cell populations, these contain disparate cell-type proportions across modalities. Table 4.1 lists the number of cells belonging to different cell-types in each domain for sciOmics and MEC datasets.

Table 4.1: Number of cells in (and percentages of) each cell-type across different modalities in the scNMT-seq co-assayed dataset after quality control procedures and the non-coassay datasets.

	Modality #1 (Gene Expression)	**Modality #2** (Chromatin Accessibility)	**Modality #3** (DNA Methylation or 3D chromosomal conform.)
scNMT dataset	**(n = 579)** **E4.5:** 76 (12.73%) **E5.5:** 104 (17.42%) **Day6.5:** 146 (24.46%) **E7.5:** 271 (45.39%)	**(n = 647)** **E4.5:** 63 (9.73%) **E5.5:** 89 (13.76%) **E6.5:** 220 (34.00%) **E7.5:** 175 (42.50%)	**(n =725)** **E4.5:** 65 (8.96%) **E5.5:** 91 (12.55%) **E6.5:** 278 (38.34 %) **E7.5:** 291 (40.14%)
sciOmics dataset	**(n = 1,058)** **Day0:** 489 (46.22%) **Day3:** 127 (12.00%) **Day7:** 78 (7.37%) **Day11:** 145 (13.71%) **NPC:** 219 (20.70%)	**(n = 1,296)** **Day0:** 164 (12.65%) **Day3:** 702 (54.17%) **Day7:** 77 (5.94%) **Day11:** 175 (13.50%) **NPC:** 178 (13.73%)	**(n =2,154)** **Day0:** 987 (45.82 %) **Day3:** 435 (20.19 %) **Day7:** 243 (11.28 %) **Day11:** 164 (7.61 %) **NPC:** 325 (15.09 %)
MEC dataset	**(n=26,273)** **Basal:** 11,138 (42.39 %) **L-Sec (Prog):** 7,683 (29.24 %) **L-HR:** 3,439 (13.09 %) **L-Sec (Mat):** 2,869 (10.92 %) **L-Sec (Prolif):** 758 (2.89 %) **Stroma:** 386 (1.47 %)	**(n=21,262)** **Basal:** 13,353 (62.80 %) **L-Sec (Prog):** 3,343 (15.72 %) **L-HR:** 2,624 (12.34 %) **L-Sec (Mat):** 1,165 (5.48 %) **L-Sec (Prolif):** 7 (0.033 %) **Stroma:** 770 (3.62 %)	N/A

4.4.2 Evaluation metrics and baseline methods

Although most of the datasets lack 1–1 cell correspondences, we can evaluate alignment using cell-type labels through label transfer accuracy (LTA) as in [13, 14, 25, 28]. This metric assesses the clustering of cell-types after alignment by training a kNN classifier on a training set (50% of the aligned data) and then evaluates its predictive accuracy on a test dataset (the other 50% of the aligned data). Higher values correspond to better alignments, indicating that cells that belong to the same cell-type are aligned close together after integration. We benchmark our method against the current unsupervised single-cell multi-omic alignment methods, Pamona [14], UnionCom [13], MMD-MA [79], bindSC [35], Seuratv4 [83], and the previous version of SCOT, which performs alignment without the KL term [25, 28]. Pamona [14], as previously

discussed, uses partial Gromov-Wasserstein (GW) optimal transport to align single-cell datasets. UnionCom [13] performs unsupervised topological alignment through a two-step procedure that first finds a correspondence between the domains, considering both global and local geometries with a hyperparameter to control the trade-off between them, and then embeds them in a new shared space. MMD-MA [79] uses the maximum mean discrepancy (MMD) measure to align and embed two datasets in a new space. BindSC [35] requires the users to bring input datasets to the gene expression feature space by constructing a gene activity score matrix for the epigenomic domains, then finds a correspondence matrix between samples through bi-order canonical correspondence analysis (bi-CCA), and jointly embeds the domains into a new space. Finally, Seuratv4 [83] also requires gene activity score matrices for epigenomic domains and then identifies correspondence anchors via CCA. Based on these anchors, it imputes one genomic domain based from the other domain and co-embeds them into a shared space using UMAP.

Since bindSC and Seurat v4 require the creation of gene activity score matrices for epigenomic datasets, they might be more difficult to use with certain sequencing domains. For instance, gene activity scoring is challenging for 3D chromosomal conformation. Of all the selected baselines, only Pamona and UnionCom can align more than two domains, so we only use them as baselines for experiments with multiple domains ($M > 2$). For each benchmark, we define a hyperparameter grid of similar granularity and perform extensive tuning (see Section 4.5). We report the alignment results with the best performing hyperparameter combinations in Section 4.6.1.

4.5 Hyperparameter Tuning Procedure Details

For each alignment method, we define a grid of hyperparameters and choose the best performing combination for each experiment. If methods share similar hyperparameters in their formulation, we keep the range defined for these consistent across

all algorithms. Examples for such hyperparameters are dimensionality of the latent space, p, for the algorithms that commonly embed datasets; entropic regularization constant, ϵ, for methods that employ optimal transport; number of neighbors, k, for methods that model single-cell datasets with nearest neighbor graphs. Otherwise, we refer to the publication and the code repository for each method to choose a hyperparameter range.

For Pamona, we tune four hyperparameters: $k \in \{20, 30, \ldots, 150\}$, the number of neighbors in the cell neighborhood graphs, $\epsilon \in \{5e-4, 3e-4, 1e-4, 7e-3, 5e-3, \ldots, 1e-2\}$, the entropic regularization coefficient for the optimal transport formulation, $\lambda \in \{0.1, 0.5, 1, 5, 10\}$, the coefficient for the trade-off between aligning corresponding cells and preserving local geometries, and lastly, $p \in \{3, 4, 5, 10, 30, 32\}$, the output dimension for embedding. We choose the ranges for ϵ and k to be consistent with the corresponding hyperparameters in SCOT and SCOTv2 algorithms and the ranges for the embedding dimensions to be consistent with the recommended values in MMD-MA and UnionCom embeddings.

For UnionCom, we tune the trade-off parameter $\beta \in \{0.1, 1, 5, 10, 15, 20\}$ and the regularization coefficient $\rho \in \{0, 0.1, 1, 5, 10, 15, 20\}$ based on the ranges reported by Cao *et al.* in the publication [13]. We additionally tune the maximum neighborhood size permitted in the neighborhood graphs, $k_{max} \in \{40, 100, 150\}$, as well as the embedding dimensionality $p \in \{3, 4, 5, 10, 30, 32\}$. The sweep range for hyperparameter k_{max} is smaller than the other hyperparameters because UnionCom automatically starts from $k = 2$ and goes up to k_{max} to find the lowest k that returns a connected graph to use in the algorithm. Therefore, more refined search is not needed.

For MMD-MA, we choose the weights λ_1 and $\lambda_2 \in \{1e-2, 5e-3, 1e-3, 5e-4, \ldots, 1e-9\}$. This range includes the hyperparameter range suggested by Singh *et al* ($\lambda_1,\ \lambda_2 \in \{1e-3, 1e-4, 1e-5, 1e-6, 1e-7\}$) but extends it further to increase the granularity for the sake of more fair comparison against methods that

require a higher number of hyperparameters to test, such as Pamona and UnionCom. Similarly to other methods, we also select the embedding dimensionality from $p \in \{3, 4, 5, 10, 30, 32\}$.

For bindSC, we choose the couple coefficient that assigns weight to the initial gene activity matrix $\alpha \in \{0, 0.1, 0.2, \ldots 0.9\}$ and the couple coefficient that assigns weight factor to multi-objective function $\lambda \in \{0.1, 0.2, \ldots, 0.9\}$. Additionally, we choose the number of canonical vectors for the embdedding space $K \in \{3, 4, 5, 10, 30, 32\}$.

Lastly, for Seurat v4, we tune the number of neighbors to consider when finding anchors, $k \in \{5, 10, 15, 20\}$, dimensions of the final co-embedding space, $p \in \{3, 4, 5, 10, 30, 32\}$ and the choice of the reference and anchor domains when finding anchors.

4.6 Results

4.6.1 SCOTv2 gives high-quality alignments consistently across all single-datasets

We first present the alignment results for real-world co-assay datasets with simulated cell-type imbalance. We present the results obtained by the best performing hyperparameter combinations for all methods compared in this study. Figure 4.4 (A) visualizes the barycentric projection alignments performed by SCOTv2 plotted as 2D PCA for SNARE-seq and scGEM datasets, respectively. We use barycentric projection for visualization purposes for the ease of comparison with the original domains, plotted in Supplementary Figure **??**. Here, we integrate datasets under three different settings described in the previous section: (1) Balanced datasets (or "full datasets" with no subsampling), (2) Missing cell-type in the epigenomic domains, and (3) Subsampled cells in both domains (one cell-type at 50% in the epigenomic domains and another cell-type at 75% in the gene expression domains). We include alignment results on the full datasets with 1–1 sample correspondences to ensure that

SCOTv2 performs well for balanced cases as well.

Qualitatively, we see that SCOTv2 preserves the cell-type annotations after alignment for all three settings. In Figure 4.4 (B), we report the quantitative performance of SCOTv2 and all the other state-of-the-art baselines using the Label Transfer Accuracy (LTA) scores. MMD-MA, UnionCom, Seurat, and bindSC fail to reliably align datasets with disproportionate cell-type representation across modalities. While Pamona tends to yield high-quality alignments for cases with cell-type disproportion, it fails to perform well on the SNARE-seq balanced dataset as well as its subsampling simulation. We additionally apply Pamona to randomly downsampled co-assays (Figure 4.6). We show that while Pamona's partial optimal transport framework handles cell-type disproportion better than the balanced optimal transport formulation (demonstrated by SCOT), SCOTv2 still shows an advantage in all SNARE-seq simulations ($\sim 20\%$ increase in LTA), as well as the smaller downsampling schemes ($\sim 10\%$).

Among all methods tested, SCOTv2 consistently gives more high-quality alignments across different scenarios of cell-type representation. It also demonstrates a $\sim 22\%$ average increase in LTA over the previous version of the algorithm (SCOT) when comparing the barycentric projection results and $\sim 27\%$ for the embedding results. Supplementary Figure 4.6 presents similar results (SCOTv2 attains an LTA of 0.786 followed by Pamona at 0.62 on SNAREseq and 0.542 followed by Pamona at 0.538 on scGEM) for missing cell-types in the other (gene expression) domain, suggesting that our choice of domain with missing cell-type does not affect the performance comparison results. UnionCom, Pamona, and SCOTv2 allow us to perform both barycentric projections and embed the single-cell domains in a lower-dimensional space. Overall, we observe that embedding yields higher LTA values than barycentric projection. Since the barycentric projection projects one domain onto another, the

	SNARE (full dataset)	**SNARE** (missing cell-type)	**SNARE** (subsam. dataset)	**scGEM** (full dataset)	**scGEM** (missing cell-type)	**scGEM** (subsam. dataset)	**scNMT**	**sciOmics**	**MEC**
SCOTv2	0.826	**0.653**	**0.751**	**0.509**	**0.521**	**0.415**	**0.727**	**0.537**	**0.584**
SCOT	**0.852**	0.572	0.588	0.423	0.323	0.314	N/A	N/A	0.466
Pamona	0.554	0.423	0.419	0.385	0.414	0.308	0.588	0.329	0.417
MMD-MA	0.523	0.407	0.431	0.360	0.296	0.287	N/A	N/A	0.233
UnionCom	0.411	0.406	0.422	0.332	0.315	0.276	0.474	0.306	0.349
bindSC	0.713	0.584	0.475	0.387	0.254	0.262	N/A	N/A	0.412
Seurat	0.428	0.517	0.503	0.408	0.377	0.329	N/A	N/A	0.387

Table 4.2: Alignment performance benchmarking in the fully unsupervised setting. We run SCOTv2 and SCOT using their heuristics to approximately self-tune hyperparameters. We use default parameters for other methods due to a lack of similar procedures for unsupervised self-tuning.

separation of the domain being projected onto (or anchor domain) limits the clustering separation after alignment. In contrast, the embedding utilizes t-SNE to enhance cell-type separation, allowing for better-separated clusters after alignment.

Next, we report the alignment performance of SCOTv2 on single-cell datasets with disparities in cell-type representation due to sampling during experiments. We include scNMT, a co-assay with varying levels of cells across domains due to quality control procedures, along with sciOmics and MEC for this experiment. Note that scNMT and sciOmics have three different modalities, and hence, we can only report the baselines for methods that can align datasets with $M > 2$. Figure 4.5(A) presents the qualitative alignment results for SCOTv2 with PCA. SCOTv2 performs well on all three datasets, including the ones with three modalities. The LTA scores in Figure 4.5(B) demonstrate that SCOTv2 consistently yields the best alignments on the three real-world datasets. These results highlight its ability to reliably integrate separately sampled with disproportionate cell-type representation and multiple ($M > 2$) modalities simultaneously.

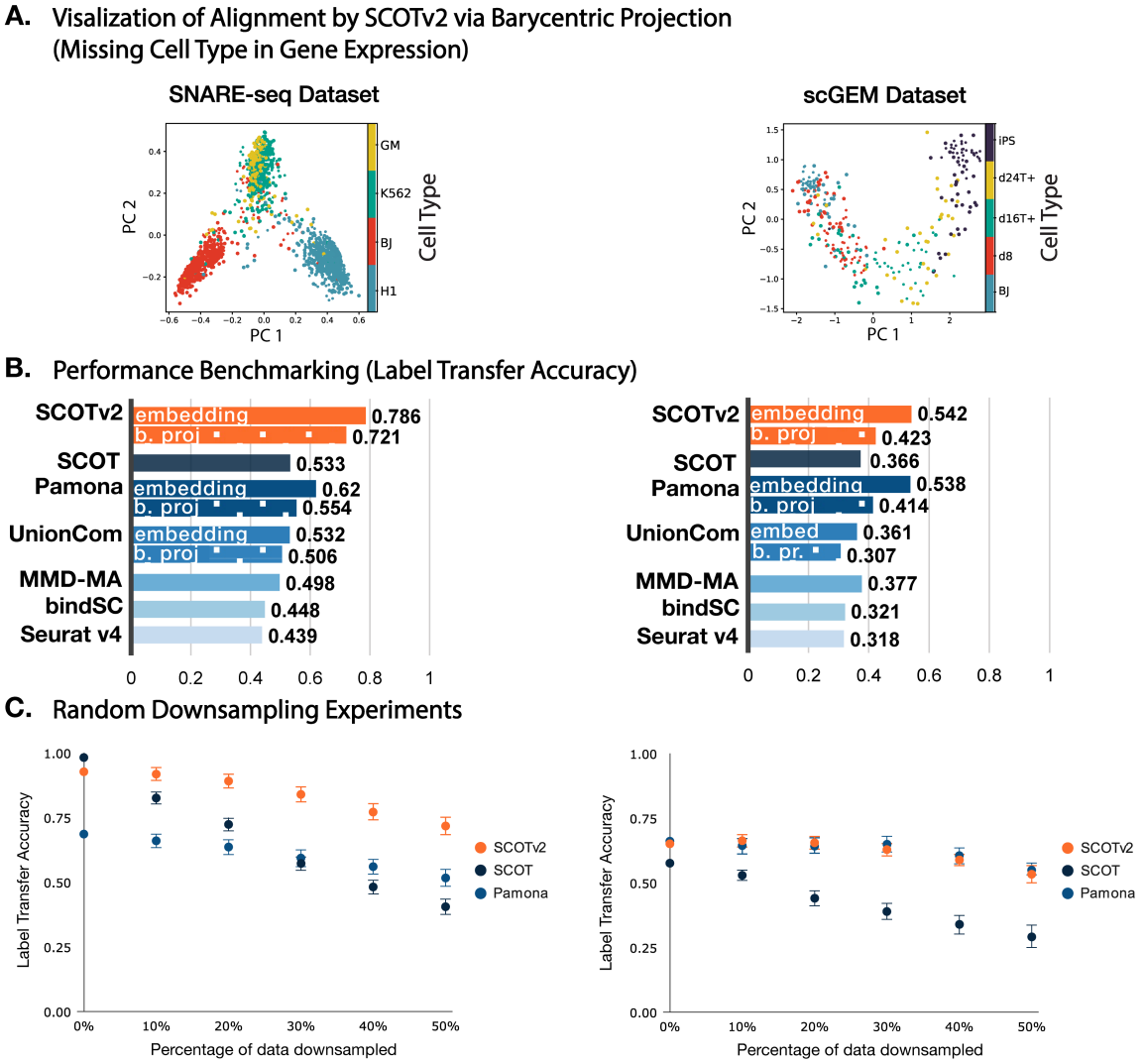

Figure 4.6: Alignment results on simulations with co-assay datasets. **A** visualizes the alignment results by SCOTv2, using barycentric projection, on co-assay datasets SNARE-seq and scGEM when a cell-type is missing in the gene expression domain. **B** quantifies the alignment quality in this experiment by using the label transfer accuracy metric and compares to baseline methods. **C** plots the average label transfer accuracy results obtained from SCOTv2, SCOT, and Pamona algorithms when aligning randomly downsampled datasets. These experiments are repeated five times and the standard deviation is shown with error bars.

4.6.2 Hyperparameter self-tuning aligns well without depending on orthogonal correspondence information

The benchmarking results above present the alignment performance of each algorithm at its best hyperparameter setting; however, users may not have 1—1 correspondences to validate alignments, for the purpose of hyperparameter selection, in real-world applications. While users may have access to cell-type labels, inferring cell-types is highly difficult in specific modalities of single-cell sequencing, such as 3D chromatin conformation. Additionally, different sequencing modalities might disagree on cell-type clustering (as is often the case with scRNA-seq and scATAC-seq datasets). In these situations, users might not have sufficient validation data for tuning hyperparameters.

We design a heuristic process (described in Section 4.3.5), as done previously for SCOT, that allows SCOTv2 to select hyperparameters in a completely unsupervised manner. Other alignment methods do not provide an unsupervised hyperparameter tuning procedure. Therefore, without validation data, a user would have to use the default parameters. In Table 4.2, we compare alignment performance for our heuristic against the default parameters of other methods. While our heuristic does not always yield the optimal hyperparameter combination, it does give more favorable results over the default settings of the other methods. Thus, we recommend using it in cases that lack orthogonal information for hyperparameter tuning.

4.6.3 SCOTv2 scales well with increasing number of samples

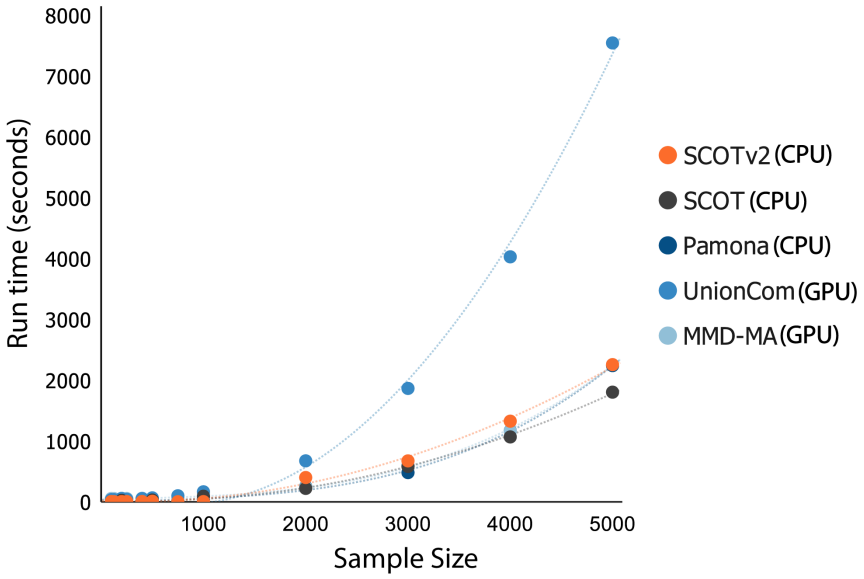

Figure 4.7: Runtimes for SCOTv2, SCOT, Pamona, UnionCom, and MMD-MA as the number of samples increases.

We compare the runtime of SCOTv2 with the top performing methods: Pamona, MMD-MA, UnionCom, and the previous version of SCOT by subsampling various numbers of cells from the MEC dataset. MMD-MA, UnionCom, and SCOTv2 have GPU versions, while Pamona and SCOT only have CPU versions. We run MMD-MA and UnionCom on a single NVIDIA GTX 1080ti GPU with VRAM of 11GB and Pamona and SCOT on Intel Xeon e5-2670 CPU with 16GB memory. We also run SCOTv2 on the same CPU to give comparable results to Pamona's runtimes. Figure 4.7 depicts that SCOT, MMD-MA, Pamona, and SCOTv2 show similar computational scaling.

4.7 Discussion

We present SCOTv2, an improved unsupervised alignment algorithm for single-cell multi-omic alignment. It extends the alignment capabilities of SCOT to datasets with

cell-type representation disproportions across different sequencing measurements. It also performs alignment for single-cell datasets with more than two measurements ($M > 2$). Experiments on real-world subsampled co-assay datasets and separately sampled and sequenced single-cell datasets demonstrate that SCOTv2 reliably yields high-quality alignments for a wide range of cell-type disproportions without compromising its computational scalability. Furthermore, SCOTv2's flexible marginal constraints enable it to consistently give good alignments results for both balanced and unbalanced single-cell datasets. In addition to effectively handling cell-type imbalances and multi-omics alignment, SCOTv2 can self-tune its hyperparameters making it applicable in complete unsupervised settings. Therefore, SCOTv2 offers a convenient way to align multiple single-cell measurements without requiring any orthogonal correspondence information.

In this second iteration of SCOT, we have utilized the coupling matrix in a new way to find a latent embedding space. While this dimension reduction improves cell-type separation, using the coupling matrix directly may offer even more insights into interactions between the aligned domains. Future work could consider how to use the probabilities in the coupling matrix directly for downstream analysis like improved clustering and pseudo-time inference. Though SCOTv2 has runtimes that scale with other methods, it requires $O(n^2)$ memory storage for the distance matrices, which may be an issue for especially large datasets. One way to address this limitation would be to develop a procedure to align a representative subset of each domain that can be extended to the entire dataset. Another way could be to co-embed datasets through a coupled autoencoder scheme and align cells in this embedding space via unbalanced optimal transport, without the use of pairwise distances. Therefore, future work could explore these directions to further improve the scalability of SCOTv2.

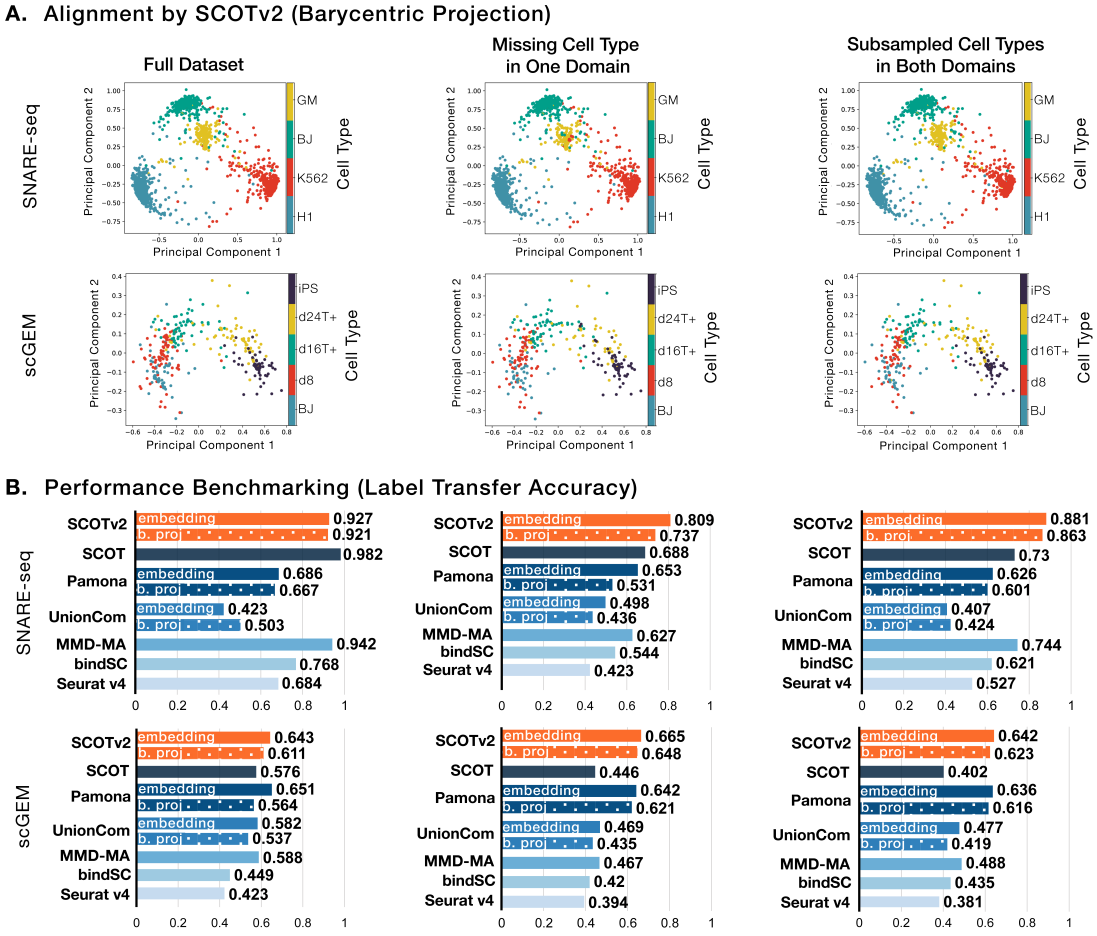

Figure 4.4: Alignment results for simulations and balanced co-assay datasets. **A** visualizes the barycentric projection alignment on SNARE-seq and scGEM for the full co-assay datasets, simulations with a missing cell-type in the epigenomic domain, and subsampled cell-types in both domains. **B** compares the alignment performance of SCOTv2 to the benchmarks through LTA. For SCOTvs, Pamona, and UnionCom, we report results on both embedding into a shared space (solid bars) and the barycentric projection (dotted bars).

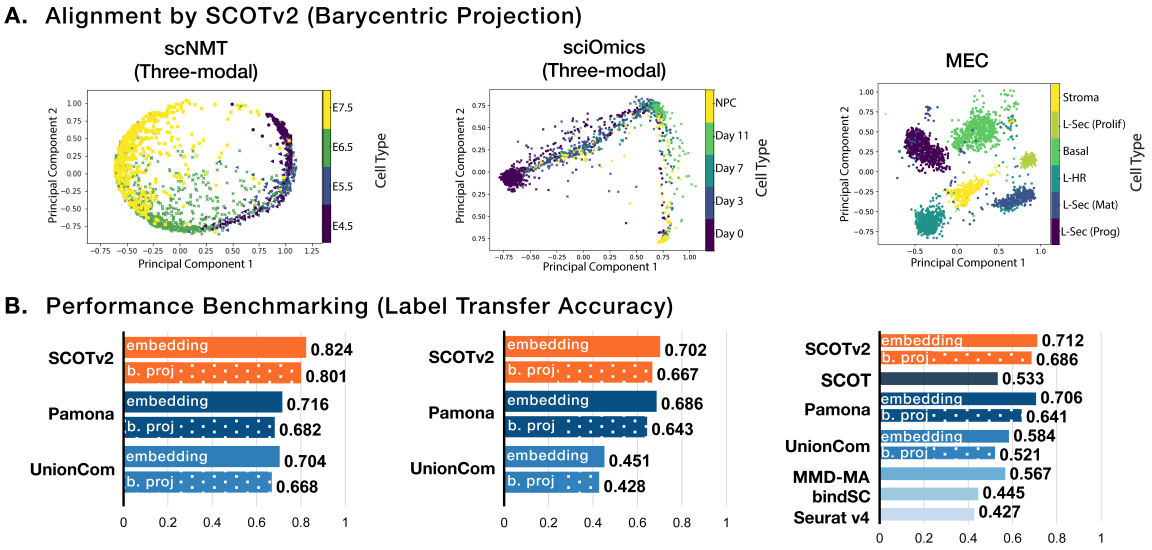

Figure 4.5: Alignment results for multi-modal ($M > 2$) and separately sequenced datasets. **A** visualizes the alignment of scNMT-seq, sciOmics, and MEC. All datasets have unequal sample sizes and cell-type proportions across domains. **B** benchmarks alignment performance through LTA. As in Figure 4.4, we report results both by embedding (solid bars) and barycentric projection (dotted bars) for the methods that allow for both. For scNMT-seq and sciOmics, which are three-modal datasets, we only demonstrate results for SCOTv2, Pamona, and UnionCom, which can handle more than two modalities.

Chapter 5

Jointly aligning samples and features of datasets

5.1 Introduction

As introduced in Chapter 1.1, a large motivation behind obtaining paired or integrated measurements of single-cell multi-omics is to study the rules of cell regulation by combining information about different genomic events. Although the algorithms presented so far in Chapters 3 and 4 align multi-omic measurements from single-cell experiments, they do not reveal possible relationships between the genomic features captured in these measurements. Given the size of the feature space in typical sequencing experiments (e.g. hundereds of proteins, tens of thousands of genes, millions of chromatin regions), generating ranked hypotheses on potential cross-modal relationships could help biologists with prioritizing experiments. Here, we present a novel algorithm, "Single-cell fused Gromov Co-optimal Transport (SCOOTR)", that jointly aligns both the cells and the features from unpaired single-cell multi-omic datasets. Our proposed method is also capable of leveraging supervising information from either level (on feature-feature or cell-cell relationships) in order to improve alignment quality on both levels.

5.1.1 Related Works

All existing single-cell multi-omic alignment algorithms, as presented in the previous chapters and used in the benchmarking of the SCOT and SCOTv2 algorithms, are solely capable of aligning cells. A nuanced exception to this trend is bindSC, which was released in 2022 after the publications of SCOT and SCOTv2. The bindSC method uses an alternating optimization procedure, using bi-order cannonical correlation analysis, to align cells while also aligning features. Although bindSC is released as a cell-cell alignment method, it is possible to access to the feature alignments it computes during its optimization. Therefore, in this study, we use bindSC as a baseline while evaluating our feature-level alignments.

5.1.2 Our contributions

- We propose a new optimal transport formulation,**S**ingle-cell fused gromov **CO**-**O**ptimal **TR**ansport (SCOOTR), that interpolates between Gromov-Wasserstein distance and co-optimal transport . As a result of this interpolation, our proposed divergence can compare probability measures or datasets across different metric spaces, while leveraging both structural information related to the underlying data geometries and the raw features.

- Our method jointly solves for two coupling matrices, one that relates samples across the datasets, and one that relates features (Figure 5.1).

- We demonstrate that the new formulation gives improved cell alignments compared to SCOT and SCOTv2.

- We demonstrate that users can leverage any prior information that exists on either the feature or the sample (i.e. cell) level relationships in order to improve the quality of the coupling matrices recovered.

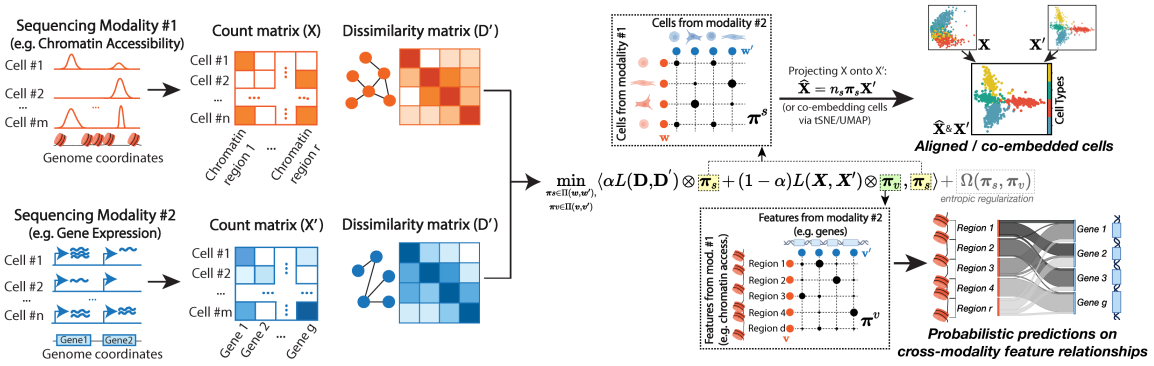

Figure 5.1: Schematic outlining the SCOOTR algorithm. Given two single-cell multi-omic datasets, SCOOTR seeks to find two coupling matrices, one aligning cells, and the other aligning features across these datasets.

5.2 Method

Notations In what follows, we denote by $\Delta_n = \{\mathbf{w} \in (\mathbb{R}_+)^n : \sum_{i=1}^n w_i = 1\}$ the simplex histogram with n bins. We use $\otimes$ for tensor-matrix multiplication, *i.e.*, for a tensor $\mathbf{L} = (L_{i,j,k,l})$, the tensor-matrix multiplication $\mathbf{L} \otimes \mathbf{B}$ is the matrix $(\sum_{k,l} L_{i,j,k,l} B_{k,l})_{i,j}$. We use $\langle \cdot, \cdot \rangle$ for the matrix scalar product associated with the Frobenius norm $\| \cdot \|_F$. Finally, we write $\mathbb{1}_d \in \mathbb{R}^d$ for a d-dimensional vector of ones and denote all matrices by upper-case bold letters (*i.e.*, $\mathbf{X}$) or upper-case Greek letters (*i.e.*, Γ); all vectors are written in lower-case bold (*i.e.*, $\mathbf{x}$). We use the terms "coupling matrix" and "correspondence matrix" interchangeably.

Monge-Kantorovich problem Let $\mathbb{X}, \mathbb{Y}$ be two subsets of $\mathbb{R}^d$ and $c : \mathbb{X} \times \mathbb{Y} \to \mathbb{R}_{\geq 0}$ be a lower semi-continuous cost function defined for all $(\mathbf{x}, \mathbf{y}) \in \mathbb{X} \times \mathbb{Y}$. Given two probability measures $\mu \in \mathcal{P}(\mathbb{X}), \nu \in \mathcal{P}(\mathbb{Y})$, where MK seeks a **coupling** $\gamma \in \Pi(\mu, \nu)$ minimizing the following quantity:

$$W_c(\mu, \nu) = \mathbb{E}_{\gamma \in \Pi(\mu,\nu)} c(\mathbf{x}, \mathbf{y}), \tag{5.1}$$

where $\Pi(\mu, \mu)$ is the space of probability distributions over $\mathbb{R}^2$ with marginals μ and ν. Such optimization problem defines a proper metric on the space of probability distributions called the Wasserstein distance.

In the discrete version of the problem, one deals with the empirical measures supported on vectors $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n \in \mathbb{X}$ and $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^m \in \mathbb{Y}$ as follows:

$$\mathbf{w} = \frac{1}{n}\sum_{i=1}^{n} \delta_{\mathbf{x}_i}, \text{ and } \mathbf{w}' = \frac{1}{m}\sum_{i=1}^{m} \delta_{\mathbf{y}_i},$$

where δ_x are Dirac measures located at $\mathbf{x}$. In this case, $\Pi(p, q)$ denotes the polytope of matrices $\mathbf{\Gamma}$ such that $\mathbf{\Gamma}1 = \mathbb{1}_n, \mathbf{\Gamma}^T 1 = \mathbb{1}_m$ and (5.1) reads:

$$W_{\mathbf{C}}(\mathbf{w}, \mathbf{w}') = \min_{\mathbf{\Gamma} \in \Pi(\mathbb{1}_n, \mathbb{1}_m)} \langle \mathbf{\Gamma}, \mathbf{C} \rangle_F, \tag{5.2}$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product, $\mathbf{C} \geq 0$ is a cost matrix $\in \mathbb{R}^{n_1 \times n_2}$, representing the pairwise transportation costs. One can also add an entropic regularization $E(\mathbf{\Gamma}) := \sum_{ij} \mathbf{\Gamma}_{ij}(\log \mathbf{\Gamma}_{ij} - 1)$ to (5.2) to obtain a strongly convex optimization problem with smoother solutions that can be obtained using a simple matrix balancing algorithm.

Gromov-Wasserstein distance When the input spaces $\mathbb{X}, \mathbb{Y}$ are different, for instance, $\mathbb{X}, \mathbb{Y}$ are subsets of $\mathbb{R}^d$ and $\mathbb{R}^{d'}$, respectively, one cannot use the above-mentioned OT formulations as they rely on a cost function c that is defined only for comparable metric spaces. To circumvent this limitation, a new OT distance between two metric-measure spaces termed **Gromov-Wasserstein** (GW) [61] was proposed.

The idea behind GW problem is as follows: instead of aligning vectors from $\mathbf{X}$ and $\mathbf{Y}$, we align their pairwise distances (or similarities) by assuming that coupling values should be higher for pairs of points that exhibit higher intra-domain similarities. More formally, given two metrics $d_{\mathbb{X}} : \mathbb{X} \times \mathbb{X} \to \mathbb{R}_+$ and $d_{\mathbb{Y}} : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}_+$, we first define two

matrices $\mathbf{D^X} \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$ and $\mathbf{D^Y} \in \mathbb{R}^{\mathbf{m}\times\mathbf{m}}$ such that $(\mathbf{D^X})_{\mathbf{ij}} = \mathbf{d}_{\mathbb{X}}(\mathbf{x_i}, \mathbf{x_j})$ and similarly for $\mathbf{D^Y}$. The GW distance for measure metric spaces $(\mathbf{w}, d_{\mathbb{X}})$ and $(\mathbf{w'}, d_{\mathbb{Y}})$ in this case is defined as follows:

$$\mathrm{GW}(\mathbf{D^X}, \mathbf{D^Y}, \mathbf{w}, \mathbf{w'}) := \min_{\Gamma\in\Pi(\mathbf{w},\mathbf{w'})} \sum_{i,j,k,l} L(\mathbf{D}^X_{i,k}, \mathbf{D}^X_{j,l})\mathbf{\Gamma}_{i,j}\mathbf{\Gamma}_{k,l} = \langle \mathbf{L}(\mathbf{D^X}, \mathbf{D^Y}) \otimes \Gamma, \Gamma\rangle_F, \tag{5.3}$$

where $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ is an arbitrary loss function, usually the quadratic-loss or Kullback-Leibler divergence. Similarly, one can define an entropic version of the Gromov-Wasserstein distance as done in [69].

COOT follows a different idea and aims to align $\mathbf{X}$ and $\mathbf{Y}$ in their original space by solving:

$$\mathrm{COOT}(\mathbf{X}, \mathbf{Y}, \mathbf{w}, \mathbf{w'}, \mathbf{v}, \mathbf{v'}) := \min_{\Gamma^s\in\Pi(\mathbf{w},\mathbf{w'}),\Gamma^v\in\Pi(\mathbf{v},\mathbf{v'})} \quad \sum_{i,j,k,l} L(\mathbf{X}_{i,k}, \mathbf{Y}_{j,l})\Gamma^s_{i,j}\Gamma^v_{k,l}, \tag{5.4}$$

where $\mathbf{v}$ and $\mathbf{v'}$ are empirical distributions associated with the features (columns) of $\mathbf{X}$ and $\mathbf{Y}$.

5.2.1 Single-cell fused Gromov Co-Optimal Transport (SCOOTR)

While Gromov-Wasserstein optimal transport accounts for dataset structure during the alignment of samples, it discards features. Co-optimal transport (COOT), on the other hand, jointly aligns both samples and features, but does not leaverage the information on dataset geometries given by pairwise distances, which proved to be beneficial in single-cell dataset integration tasks, as demonstrated by SCOT and SCOTv2 algorithms [25, 27, 29]. We propose an interpolation between the two to

combine the best of both worlds:

$$\min_{\substack{\Gamma^s \in \Pi(\mathbf{w}, \mathbf{w}'), \\ \Gamma^v \in \Pi(\mathbf{v}, \mathbf{v}')}} \langle \alpha L(\mathbf{D}^{\mathbf{X}}, \mathbf{D}^{\mathbf{Y}}) \otimes \Gamma^s + (1-\alpha) L(\mathbf{X}, \mathbf{Y}) \otimes \Gamma^v, \Gamma^s \rangle. \tag{5.5}$$

with $L = L_2$. Similarly to SCOTv2, an unbalanced formulation can be obtained with the addition of the KLD terms over the coupling marginals. We additionally add the entropic regularization term for both coupling matrices $\epsilon_1 H(\Gamma^s) + \epsilon_2 H(\Gamma^v)$ in order to make computation empirically more efficient and employ Sinkhorn iterations in optimization. The choice of α hyperparameter determines the strength of interpolation between Gromov-Wasserstein distance and COOT. We show in the proofs in Section 7.1 that as α approaches 1, we retrieve Gromov-Wasserstein distance, and as α approaches 0, SCOOTR behaves like COOT.

5.2.2 Providing supervision to SCOOTR

A useful property of SCOOTR that is inherited from COOT is that one can provide a weak supervision when solving for $\mathbf{\Gamma}^s$ and $\mathbf{\Gamma}^v$ by scaling the costs of matching samples/features that should not be matched together. We do this by multiplying the cost matrix $\boldsymbol{L}$ with a supervision matrix $\boldsymbol{D}$ that user provides in inputs. This matrix can be provided for either the sample or the feature alignments. If it's provided for the feature-level alignments, for example, it is only used in the optimization step for the feature-feature coupling matrix in the block coordinate descent (Supplementary Algorithm 7). The entries of the supervision matrix are expected to range between 0 and 1. For example, an entry for 0 for the row i and column j in the feature-level supervision matrix removes the cost associated with aligning the feature i and feature j of the two input datasets, respectively. We will show that this unique feature of

SCOOTR is valuable when dealing with single-cell measurements.

Algorithm 7: Pseudocode for SCOOTR

Input: $\boldsymbol{X}, \boldsymbol{X'}, \boldsymbol{D^X}, \boldsymbol{D^{X'}}, \alpha, \epsilon_1, \epsilon_2, \boldsymbol{M}_s$(optional),$\boldsymbol{M}_v$(optional), choice of barycentricProjection or embedding

Initialize:

$\boldsymbol{w} \leftarrow$, Uniform$(1/n_s)$,$\boldsymbol{w'} \leftarrow$, Uniform$(1/n'_s)$,

$\boldsymbol{v} \leftarrow$, Uniform$(1/n_f)$,$\boldsymbol{v'} \leftarrow$ Uniform$(1/n'_v)$,

$\boldsymbol{\Gamma}_0^s \leftarrow \boldsymbol{w}\boldsymbol{w'}^T, \boldsymbol{\Gamma}_0^v \leftarrow \boldsymbol{v}\boldsymbol{v'^T}, t \leftarrow 0$

while $t <$ maxIter **and** $err > 0$, **do**:

  $\boldsymbol{L}_t^s \leftarrow \alpha \sum_{i,j,k,l} ||D^X_{\cdot,ik} - D^{X'}_{\cdot,jl}||^2 + (1-\alpha) \sum_{i,j} ||X_{\cdot,k} - X'_{\cdot,l}||^2 (\Gamma_t^v)_{i,j}$ // Calculating the new cost matrix for samples

  $\boldsymbol{L}_t^s \leftarrow \boldsymbol{M^s} \odot \boldsymbol{L}_t^s$ // Scaling if providing supervision on sample alignments

  $\boldsymbol{\Gamma}_t^s \leftarrow Sinkhorn(\boldsymbol{w}, \boldsymbol{w'}, \boldsymbol{L}_t^s, \varepsilon_1)$ // Optimizing the coupling matrix for samples

  $\boldsymbol{L}_t^v \leftarrow \sum_{m,n} ||X_{m,\cdot} - X'_{n,\cdot}||^2 (\Gamma^s_{t-1})_{m,n}$ // The new cost matrix for features

  $\boldsymbol{L}_t^v \leftarrow \boldsymbol{M^v} \odot \boldsymbol{L}_t^v$ // Scaling if providing supervision on feature alignments

  $\boldsymbol{\Gamma}_t^v \leftarrow Sinkhorn(\boldsymbol{v}, \boldsymbol{v'}, \boldsymbol{L}_t^v, \varepsilon_2)$ // Optimizing the coupling matrix for features

  err $\leftarrow ||\boldsymbol{\Gamma}_t^s - \boldsymbol{\Gamma}_{t-1}^s||^2$

  $t \leftarrow t+1$

Align cells in the same space:

if *Barycentric projection* **then**

  $\widehat{\mathbf{X}} = n_s \Gamma^s \mathbf{X}'$ and $\widehat{\mathbf{X}}' = \mathbf{X}'$

end

else

  // Find shared embedding (same as in SCOTv2)

  $\widehat{\mathbf{X}}, \widehat{\mathbf{X}} \leftarrow \min_{\widehat{\mathbf{X}}, \widehat{\mathbf{X}'}} \mathrm{KL}(P||Q') + \beta||X - X'(\Gamma^s)^T||_F^2$

end

Return: $\widehat{\mathbf{X}}, \widehat{\mathbf{X}}, \boldsymbol{\Gamma}_t^v$

5.3 Experimental Setup

We first show that SCOOTR can yield cell-cell alignment results on par with the existing single-cell multi-omic alignment methods. Then, we demonstrate its ability to also simultaneously align the features from different genomic modalities, using simulated and real-world single-cell sequencing datasets.

5.3.1 Datasets

When choosing datasets, we follow our main baselines – the existing optimal transport-based single-cell alignment methods [14, 25, 27, 29] and bindSC [35] – and curate a similar set of simulated and real-world datasets for a comparable benchmarking. We use the datasets with ground-truth information on 1-1 cell pairings for evaluating cell-level alignment performance. Similarly, we use the datasets with some prior information on feature correspondences for evaluating feature-level alignment performance.

Datasets for cell-cell alignment benchmarking We use one simulated dataset and three real-world single-cell multi-omic datasets to benchmark our cell alignment performance in the balanced scenario (i.e. no discrepancies in cell-type representation across datasets). The simulated dataset that has been generated by [25] using a single-cell RNA-seq data simulation package in R, called Splatter [106], also used in the experiments for SCOT in Chapter 3. We refer to this dataset as "Synthetic RNA". This dataset includes a simulated gene expression domain with 50 genes and 5000 cells divided across three cell-types, and another domain created by non-linearly projecting these cells onto a 500-dimensional space. As a result of its generation scheme, the dataset has ground-truth 1-1 cell correspondence information. To have ground-truth information on cell correspondences for evaluation, for the real-world datasets, we

choose three co-assay datasets which have paired measurements on the same individual cells: an scGEM dataset [19], a SNARE-seq dataset [18], and a CITE-seq dataset [81]. These first two datasets have been used by existing single-cell alignment methods, including the ones employing optimal transport [13, 14, 25, 27, 56, 79], while the last one was included in the evaluations of bindSC [35]. The scGEM dataset contains measurements on gene expression and DNA methylation states of 177 individual cells from human somatic cell population undergoing conversion to induced pluripotent stem cells (iPSCs) [19]. We accessed the pre-processed count matrices for this dataset through the MATCHER repository [1]. The SNARE-seq dataset contains gene expression and chromatin accessibility profiles of 1047 individual cells from a mixed population of four cell lines: H1(human embryonic stem cells), BJ (a fibroblast cell line), K562 (a lymphoblast cell line), and GM12878 (lymphoblastoid cells derived from blood) [18]. We access their count matrices on Gene Expression Omnibus, with the accession code GSE126074. Finally, the CITE-seq dataset has gene expression profiles and epitope abundance measurements on 25 antibodies from 30,672 cells from human bone marrow tissue [81]. The count matrices for this dataset were downloaded from the Seurat website [2]

To test alignments in the unbalanced scenario (when cell-type representation is disproportionate across datasets), we also include the subsampling simulations on the SNARE-seq and scGEM datasets, as well as the scNMT-seq dataset, from SCOTv2 [27, 29].

Datasets for feature-feature alignment benchmarking We assess feature-level alignment performance on real-world single-cell multi-omic datasets with some ground-truth correspondence information between the features. Among the three real-world datasets described above, we have ground-truth information on the CITE-seq dataset,

[1] `https://github.com/jw156605/MATCHER`

[2] `https://satijalab.org/seurat/v4.0/weighted_nearest_neighbor_analysis.html`

where we know which genes from the gene expression domain encode the 25 antibodies from the antibody abundance domain. We use these 1-1 correspondences to evaluate our feature alignments. Since we do not have such reliable ground-truth feature correspondence information for SNARE-seq and scGEM datasets, we use a novel computational tool called CellOracle [49]. This tool has been developed to infer regulatory networks jointly from single-cell chromatin accessibility and gene expression data. For the SNARE-seq dataset, we use CellOracle to construct gene regulatory networks. We take the chromosomal region of transcription factors and genomic elements that a gene is connected to in this network as its probable feature correspondences, and compare our alignments against these (more detail in Section 5.3.2). We are unaware of the existence of such tools for single-cell methylation data. As a result, we do not include scGEM dataset in our feature-level alignment performance benchmarking. Instead, we add a new real-world dataset with a need for single-cell alignment. This dataset contains unpaired single-cell gene expression profiles from mouse prefrontal cortex[8], and the bearded lizard pallium [89]. It has been curated with data from separately conducted experiments. Here, we have information on paralogous genes across the two species, as well as relevant cell types.

In addition to these three real-world sequencing datasets, we simulate a new set of multi-omic data with varying levels of sparsity in underlying feature-level correspondences. Our goal for including these simulations is to investigate the effect of correspondence sparsity on alignment performance. We follow the simulation set-up by Zhang *et al* [108], which modifies a single-cell RNA-seq simulation method, SymSim [107], to also simulate scATAC-seq count data based on a gene-chromosomal region relationship matrix [3]. We simulate 500 cells with 50 genes in the gene expression modality, and 1000 chromosomal regions in the chromatin accessibility modality. We randomly generate five gene-to-chromosomal region correspondence matrices with

[3] `https://github.com/PeterZZQ/Symsim2`

uniform 1-2 (sparse), 1-4, 1-6, 1-8, and 1-10 (dense) matches. We generate five multi-omic datasets using these ground-truth correspondence matrices.

5.3.2 Evaluation Criteria

Cell-cell alignment evaluation When evaluating cell-cell alignments, we use a metric previously used by other single-cell multi-omic integration tools [13, 14, 25, 27, 35, 56, 79] called "fraction of samples closer than the true match" (FOSCTTM). For this metric, we compute the Euclidean distances between a fixed sample point and all the data points in the other domain. Then, we use these distances to compute the fraction of samples that are closer to the fixed sample than its true match, and then average these values for all the samples in both domains. This metric measures alignment error, so the lower values correspond to higher quality alignments.

Feature-feature alignment evaluation To assess feature-feature alignment performance, we investigate the accuracy of feature correspondences recovered. We mainly use three real-world datasets for this task - CITE-seq, SNARE-seq, and the cross-species scRNA-seq datasets. Due to the versatility of the genomic measurements in these datasets, we follow a different procedure for each to define "ground-truth" feature correspondences to compute the accuracy.

For the CITE-seq dataset, we expect the feature correspondences to recover the relationship between the 25 antibodies and the genes that encode them. To investigate this, we simultaneously align the cells and features of the two modalities using the 25 antibodies and 25 genes in an unsupervised manner. We compute the percentage of 25 antibodies whose strongest correspondence is their encoding gene.

For SNARE-seq dataset, we start by pre-processing the dataset to prune features. The original dataset contains 18,666 genes and 1,136,771 chromosomal regions in their respective modalities. We select the top 1000 most variable genes using the `FindVariableFeatures` function of Seurat with its default parameters. We also select

the top 2500 chromosomal regions in the chromatin accessibility domain using the `FindTopFeatures` function of Signac [84]. With these, we construct a gene regulatory network with CellOracle [49] using both domains. Then, for each gene, we identify the chromosomal regions of its regulators from the regulatory network, using the human reference genome GRCh38/hg38. We expect the genes in the gene expression modality to be matched with at least one of the chromosomal regions overlapping with each regulator's genomic coordinates, and compute the accuracy over all genes accordingly.

For the cross-species RNA-seq dataset, we expect alignments between the cell-type annotations common to the mouse and lizard datasets, namely: excitatory neurons, inhibitory neurons, microglia, OPC (Oligodendrocyte precursor cells), oligodendrocytes, and endothelial cells. For this dataset, we generate cell-label matches by averaging the rows and columns of the cell-cell alignment matrix yielded by SCOOTR based on these cell annotation labels. We compute the percentage of these six cell-type groups that match as their strongest correspondence.

5.3.3 Baselines

For the cell-cell alignment evaluation, we consider the following unsupervised single-cell multi-omic integration methods, SCOT [25], SCOTv2 [27, 29], and bindSC [35]. Among these, SCOT and SCOTv2 are optimal transport-based methods; they both use Gromow-Wasserstein (GW) optimal transport [61, 70] with different relaxations. We note that unlike other alignment methods, bindSC could be considered a weakly supervised method since it requires a gene activity matrix as an input. For feature-feature alignment benchmarking, bindSC remains our only baseline since the other integration methods only perform alignment on the cell level. Although bindSC does not return the final feature-level correspondence matrix to the user, it does return the relationship between each feature and the computer intermediary factors.

By multiplying these matrices, we are able to obtain a feature-level correspondence matrix. For all methods, we set a grid of hyperparameter combinations and choose the best performing combination for each dataset. For SCOOTR, we consider the EMD (for $\epsilon_{1,2} = 0$) and Sinkhorn algorithms for each OT subproblem, entropic regularization strength for Sinkhorn taking values in $\{10^{-5}, \ldots, 10^{4}\}$, and interpolation coefficient $\alpha = \{0.0, 0.1, 0.2, ..., 1.0\}$.

5.4 Results

We apply SCOOTR on four real-world datasets, as well as a synthetic dataset to jointly align cells and genomic features of unpaired single-cell multi-omic datasets. We compare SCOOTR's feature alignment performance to bindSC, which is the only other computational tool that can perform joint alignment of cells and features. We additionally benchmark SCOOTR's cell alignment performance against the other two single-cell multi-omic alignment algorithms introduced in this thesis, SCOT and SCOTv2, which only perform cell-cell alignment.

5.4.1 SCOOTR improves upon cell alignment performance of SCOT and SCOTv2

We benchmark SCOOTR's cell-cell alignment performance under two settings: (1) the balanced case, where there is a $1-1$ correspondence between the cells of the integrated datasets, and (2) the unbalanced case, where there is a discrepancy in the cell-type proportions across the integrated datasets. For the first case, we use the mean "fraction of samples closer than true match" (FOSCTTM) metric, which measures alignment error, and for the second, we use label transfer accuracy since FOSCTTM cannot be used without ground-truth information on $1-1$ cell correspondences. Label transfer accuracy measures cell-type alignment quality, therefore higher values are better. We use SCOTv2 instead of SCOT in the second case as a

baseline, because SCOT does not perform well in unbalanced cases, as demonstrated before. As Tables 5.1 and 5.2 show, SCOOTR improves upon the cell-type alignment performance of both SCOT and SCOTv2 and outperform bindSC.

Table 5.1: Quality of cell alignments yielded by SCOT, SCOOTR,and bindSC in the "balanced case" (no disproportionate cell-type representation across datasets), as quantified by the average FOSCTTM metric (lower values are better).

Balanced Cell Alignment Experiments (Mean FOSCTTM, lower is better)				
	Splatter Simulation (Synthetic RNA-seq)	**scGEM**	**SNARE-seq**	**CITE-seq**
SCOOTR	0.0	0.183	0.136	0.109
SCOT	7.1 e-5	0.198	0.150	0.131
bindSC	3.8 e-4	0.204	0.242	0.144

Table 5.2: Quality of cell alignments yielded by SCOT, SCOOTR,and bindSC in the "unbalanced case" (disproportionate cell-type representation across datasets), as quantified by the label transfer accuracy metric (higher values are better).

Unbalanced Cell Alignment Experiments (Label transfer accuracy, higher is better)			
	scGEM (downsampled)	**SNARE-seq (downsampled)**	**scNMT-seq (Gene Exp. + Methyl.)**
SGCOOTR	**0.473**	**0.832**	**0.762**
SCOTv2	0.415	0.751	0.741
bindSC	0.262	0.575	0.754

5.4.2 SCOOTR generates biologically meaningful hypotheses on feature correspondences

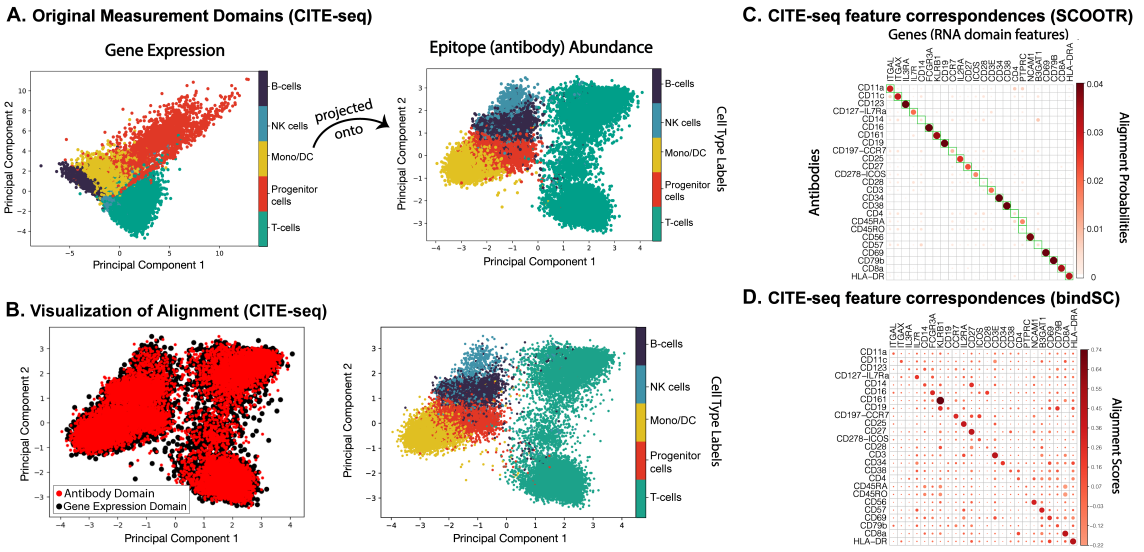

Figure 5.2: Cell-cell and feature-feature alignment results on CITE-seq dataset. **A** visualizes the original domains – antibody abundance, and gene expression, respectively –, following dimensionality reduction with 2D principal component analysis (PCA). **B** visualizes the aligned domains, after the gene expression domain has been projected onto the antibody abundance domain via barycentric projection. **C** Feature alignment probabilities recovered by SCOOTR. The green boxes along the diagonal indicate the "ground-truth" correspondences we expect to see between the antibodies and their encoding genes. **D**. The feature alignment probabilities recovered by bindSC.

Among the real-world datasets we use, CITE-seq has underlying 1-1 correspondences between the antibodies and their encoding genes. We expect SCOOTR to recover them when we align 25 antibodies with the corresponding 25 genes (while simultaneously aligning cells) in its unsupervised setting. We present the feature correspondence matrix SCOOTR yields in 5.2C. The rows and columns of this matrix are ordered such that the expected ground-truth correspondences are along the diagonal (marked by green squares). Note that the row and column probabilities add up to the weights from marginal distributions initialized in the beginning of the optimization. We observe that all antibodies are matched to their corresponding genes with a non-zero probability of correspondence and 15 of them (60%) have the strongest correspondence

probability with their encoding gene. Compared to the feature correspondence matrix we receive from bindSC (Figure 5.2D), we yield a sparser correspondence matrix while correctly aligning a higher number of antibodies with their encoding genes (compared to 13 antibody-gene pairs yielded by bindSC, giving a $\sim 52\%$ alignment accuracy). Figure 5.2B shows the cell-level alignments we receive from this run, which demonstrates that SCOOTR simultaneously recovers quality cell and feature alignments for CITE-seq dataset.

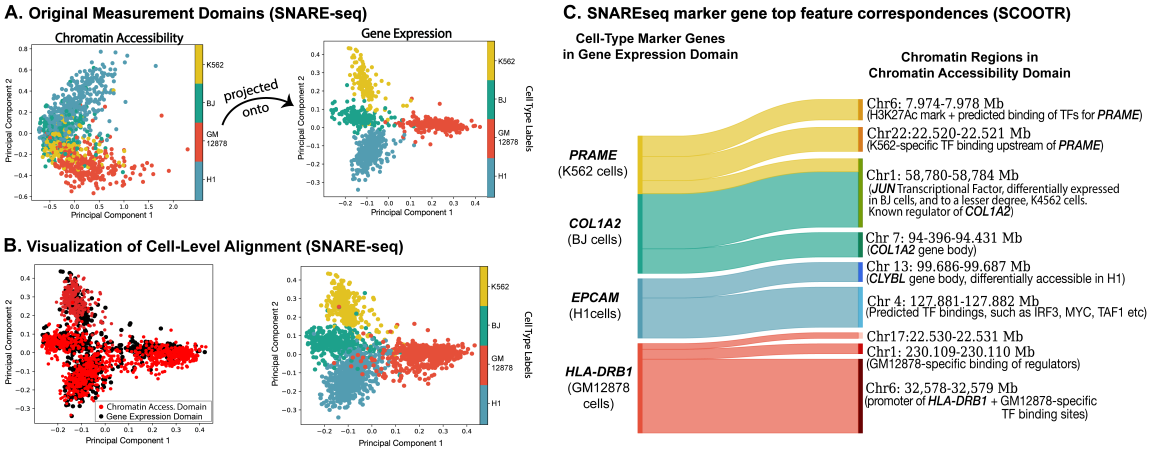

Figure 5.3: Cell and feature alignment results on SNARE-seq dataset. **A** visualizes the original domains – chromatin accessibility, and gene expression, respectively, following dimensionality reduction with 2D principal component analysis (PCA). **B** visualizes the aligned domains, after the chromatin accessibility domain has been projected onto the gene expression domain domain via barycentric projection. **C** Sankey plot visualizing the top chromatin accessibility feature correspondences recovered for the cell-type marker genes. These correspondences include the chromosomal regions of the marker genes and regions with predicted cell-type specific transcriptional factor (TF) binding.

5.4.3 Supervision on one level (e.g. cell- or feature-level alignments) improves alignment quality of both

Despite the success of unsupervised alignment on CITE-seq dataset, we observe that providing supervision on cell-type alignments greatly increases the quality of feature alignments on SNARE-seq dataset. In real-world applications, we can reasonably

expect some level of prior information to be available on the samples, namely, cell-type annotations, for at least a subset of the cells. In most cases, upon obtaining sequencing measurements, biologists use marker genes to find different cell-type groups. Although it is expected to observe some mismatch in cell type annotations between different genomic views, we demonstrate in Table 5.3 that the performance of SCOOTR improves even with weak supervision. Our unique joint alignment formulation provides the ability to perform this weak supervision at both sample and feature level. In this table, we use varying proportions $(0, 20, ...100\%)$ of the cell-type annotations to provide supervision on the cell-level to assist with the feature-level alignments. As described in the Methods section, we create a supervision matrix, which removes the cost of aligning two cells if they belong to the same cell type. In the 100% supervised setting, we use all of the cell type annotations to create this supervision matrix; whereas in the 20% supervised setting, we only use 20% of the cell type annotations. To obtain a ground-truth on correspondences, we use CellOracle [49] to infer a regulatory network using both the gene expression and the chromatin accessibility data. We expect to recover the correspondences between genes and at least one segment of the chromosomal regions corresponding to each of their regulators (described in Section 5.3.2). We visualize examples of chromatin accessibility to gene expression correspondences yielded by SCOOTR for the cell-type marker genes in Figure 5.3(C).

We look into biological annotations of the matching chromatin accessibility regions on UCSC Genome Browser [65] with annotations from JASPAR Transcription Factor Binding Site Database [15] and ReMap Atlas of Regulatory Regions [42]. We observe that the marker genes are matched with their chromosomal region or the regions associated with relevant transcription factor binding sites. For example, the strongest correspondence of *COL1A2*, the marker gene of BJ cell-line, is in the chromosomal region of *JUN*, which is a transcription factor identified to be differentially expressed in BJ cells, and to a lesser level, K562 cells [18]. Its second strongest match

is a region within its own chromosomal region. Similarly, the marker gene for the GM12878 cell line, *HLA-DRB1*, is most strongly matched with a region upstream of its own genomic region, along with predicted GM12878-specific transcriptional binding sites. Another example is *PRAME* and Chr22: 22.520-22.521 Mb region, which is a region upstream of the *PRAME* gene body that is rich with predicted transcriptional factor (TF) binding sites according to the "RepMap Atlas of Regulatory Regions" [42] annotations on UCSC Genome Browser (Human hg38 annotations) [65]. Among the predicted TF bindings, many of them are K562-specific predictions, and some of these are known regulators of *PRAME*, such as but not limited to *E2F6*, *HDAC2*, *CTCF* (based on GRNdb database [37] of TF-gene relationships). Additionally, *COL1A2* and *HLA-DRB1* also have recovered correspondences with their own chromosomal region, "Chr7:94.396-94.421 Mb" and "Chr6:32,578-32,579 Mb", respectively. We observe that *COL1A2* and *PRAME* are also additionally aligned with "Chr1: 58,780 - 58,784 Mb" regions, which correspond to the gene body of *JUN* transcriptional factor. Indeed, *JUN* has been identified as one of the transcriptional factors differentially expressed in the K562 and BJ cells, but more strongly in the latter, according to the original publication that released this dataset [18]. GRNdb also identified *JUN* to be one of the regulators of the *COL1A2* gene. In addition to the chromosomal region of *JUN*, *PRAME* has another region abundant in predicted TF binding sites among its top correspondences: "Chr6: 7.974-7.978 Mb". This region is annotated with an H3K27Ac mark on the UCSC Genome Browser, and has multiple predicted binding sites of TFs GRNdb identifies as regulators of *PRAME*, such as *IRF1, HDAC2, HOXC6* and POU2AF1. The *HLA-DRB1* gene is also aligned with a chromosomal region rich in GM12878-specific predictions of TF bindings, such as *IRF4*, *IRF8*, *ETV6*, and *CREM*, which GRNdb lists as potential regulators of *HLA-DRB1*. Lastly, even though we couldn't find a biological relationship reported in the

literature between the *CLYBL* gene and *EPCAM* gene (marker gene for the H1 cell-line), the chromosomal region in *CLYBL* body where SCOOTR finds a correspondence with *EPCAM* indeed appears to be differentially accessible in H1 cells (and in to a lesser degree, K562) in our dataset.

Table 5.3: Feature alignment performance on SNARE-seq and cross-species RNA-seq dataset with increasing supervision on cell-type alignments.

Supervision level (%) on cell-type alignments	**0 %**	**20 %**	**40 %**	**60 %**	**80 %**	**100 %**	**bindSC**
Accuracy (%) of feature alignments	31.22 %	42.91 %	54.82 %	63.48 %	71.84 %	79.67 %	40.26 %

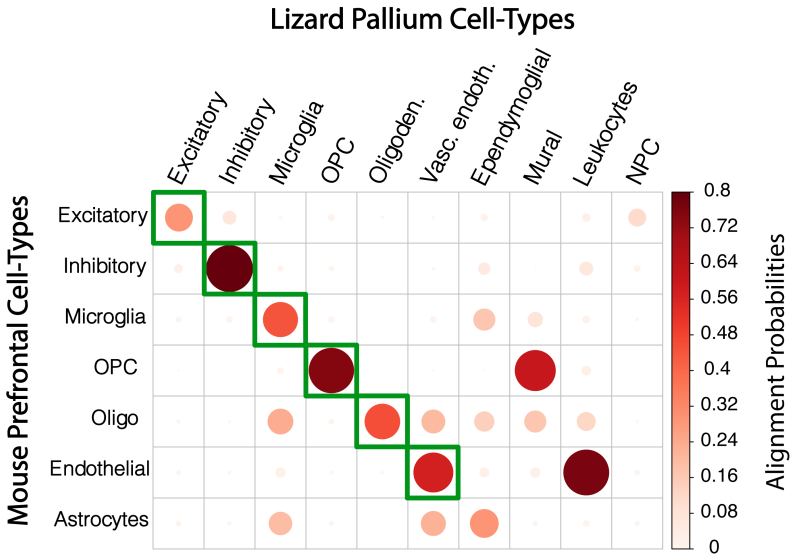

Figure 5.4: **Cell-type alignment results on cross-species dataset.** Full supervision is provided on the 10,816 paralogous genes between mice and lizard.

Similarly, we demonstrate that supervision on the feature-level alignments improves cell-level alignment quality. For this, we align the gene expression data obtained from the brain tissue of two different species, namely, mouse pre-frontal cortex and bearded dragon lizard pallium. Since these are separately profiled datasets, we do not expect to find 1-1 correspondences between the cells; however, we have prior

information on relatedness of cell-type labels (Section 5.3.2). Here, we provide feature-level supervision to guide SCOOTR to recover these cell-type relationships. Coming from different species, these datasets do not have the exact same set of features, however, they share 10,816 paralogous genes. We create a supervision matrix, similar to the case in SNARE-seq, to guide the alignments between paralogous genes across the two datasets and investigate the alignments yielded for the cells. We average the cell-cell alignment probabilities across cell types to get the cell-type alignments. Figure 5.4A demonstrates the cell-type alignments we receive from fully supervised case, and the Table 5.4 presents the cell-type alignment accuracy with varying levels $(0\%, 20\%, ..., 100\%)$ of feature alignment supervision. Here, accuracy is calculated as a percentage of correspondences recovered among all the expected correspondences (marked by the green boxes in Figure 5.4A). Similarly to cell alignment, we see that supervision on feature alignments increases cell-level alignment accuracy. So, when validation data is present on feature relationships, they can be used to obtain higher quality cell-cell alignments using SCOOTR. Additionally, this application demonstrates that SCOOTR can also be used to relate cell clusters from different datasets.

Table 5.4: Cell-type alignment performance on cross-species RNA-seq dataset with increasing supervision on paralogous gene alignments.

Supervision level (%) on feature alignments	**0 %**	**20 %**	**40 %**	**60 %**	**80 %**	**100 %**	**bindSC**
Accuracy (%) of sample alignments	66.67 %	83.34 %	83.34 %	100 %	100 %	100 %	83.34 %

5.5 Discussion

The majority of the existing single-cell multi-omic alignment methods solely align cells. Our proposed method SCOOTR jointly aligns both the cells and the features

of single-cell multi-omic datasets, allowing researchers to study potential relationships between different views of the genome. We intend SCOOTR to be a hypothesis-generation tool for biologists. The correspondence probabilities that SCOOTR yields can be used to rank the predicted cross-measurement relationships, allowing scientists to prioritize downstream investigations accordingly when studying regulatory interactions.

Single-cell alignment methods require validation data on cell-cell correspondence to tune the hyperparameters. However, such information is unlikely to be present in real-world cases when datasets are separately sequenced. Although both SCOT and SCOOTR perform self-tuning by tracing optimal transport cost, the lowest cost does not always correspond to the best alignment, and the quality of self-tuning can vary between datasets. If prior information other than cell-cell alignment validations is present –such as the paralogous genes in the case of cross-species alignment experiments or the cell-type annotations in SNARE-seq experiments–, using these could lead to better alignments in some datasets compared to self-tuning. Our experiments demonstrate that even partial supervision leads to improvement in alignment performance.

For the feature-level alignments, neural-network-based formulations of fused Gromov co-optimal transport could potentially allow one to account for more complex relationships. However, maintaining feature-level interpretability in the coupling matrix becomes more challenging in this formulation. Investigating an interpretability-preserving neural formulation remains to be a future work. Additionally, it might be possible to set the marginal distributions over cells and features based on common biological knowledge. For example, when aligning gene expression data and chromatin accessibility data, one might scale the weights of chromosomal regions corresponding to common transcription binding sites based on existing databases, guiding the algorithm to align these with more genes. Similarly, one could scale the weights of the

cells based on prior clustering without the need for cell-type annotations. Future work could compare such approaches to the unsupervised and supervised cases presented here. In the meantime, we allow users to customize marginal distributions when running SCOOTR. Overall we demonstrate that SCOOTR is a competitive single-cell multi-omic data integration method that can help generate hypotheses for genomic feature relationships when jointly studying multiple single-cell datasets.

Chapter 6

Conclusion

Versatility of single-cell sequencing technologies gives an unprecedented opportunity to study how the different genomic features co-vary across cells. These studies can give new biological insights into the mechanisms behind how the different events in the genome interact to regulate cells. Due to the experimental challenges behind single-cell multi-omic (i.e. multi-modal genomic) profiling, it is difficult to experimentally obtain multi-omic data at the single-cell resolution. Therefore, studying multi-omic relationships requires computational integration of separately profiled single-cell datasets with different genomic measurements. Despite the importance of this problem, many computational methods developed have not addressed the real-world challenges that arise when integrating these datasets.

In this dissertation, we presented three algorithms that make contributions towards addressing these challenges, such as automatically tuning hyperparameters in the absence of sufficient validation data, and accounting for disproportionate cell-type representation that commonly occurs when measurements are taken from different samples of a cell population. To our knowledge, no existing single-cell alignment algorithm has a similar attempt of self-tuning hyperparameters and, as demonstrated in Chapter 4, most algorithms fail to take into account potential cell-type representation disparities across the datasets they integrate. The second algorithm we present,

SCOTv2, which is build on SCOT, addresses both of these challenges and yields competitive results in a range of settings, including the cases where there are missing cell types or different numbers of cells from each cell type in different datasets integrated, as well as real-world unpaired and paired single-cell multi-omic datasets. In addition, the cell-cell coupling matrices yielded by the optimal transport framework we employ in SCOT and SCOTv2 have unique advantages compared to the other alignment algorithms that solely co-embed datasets in a shared space. While co-embedding visualizes which cells are most likely to be similar to each other across measurement modalities, they do not provide information on what sort of profile a cell from one measurement modality (e.g. gene expression) would look like in another measurement modality (e.g. chromatin accessibility). On the other hand, we can obtain this information via barycentric projection. Additionally, the coupling matrix gives probabilistic information on cell-cell similarities as opposed to deterministic embeddings.

SCOOTR further improves on cell-type alignment quality of SCOT and SCOTv2 by considering not only the dataset geometries, but also the relationships between features, yielding hypotheses on possible multi-omic feature interactions. We expect SCOOTR to be instrumental in discovering new biological relationships across different genomic domains, without requiring paired measurements. It can also be used to leverage any supervising information (however partial) a user may have on either cell-type or feature correspondences to improve alignment quality on both levels, as demonstrated in Chapter 5.

While both SCOTv2 and SCOOTR give promising results on single-cell multimodal integration experiments, there are a number of avenues for further improvement through future work. First of these is regarding the runtime of the algorithms. The entropically regularized Gromov-Wasserstein optimal transport has a sample complexity of $O(n^2)$, that is, it scales quadratically with the number of cells. As the

sequencing technologies get higher throughput, it is becoming possible for a single sequencing experiment to yield data on $\sim 1,000,000$ cells. A runtime complexity of $O(n^2)$ is not ideal in such a scenario. A potential way to address this issue could be to use approximate solvers for Gromov-Wasserstein optimal transport, which might yield lower quality alignments but in a faster manner, such as sliced Gromov-Wasserstein [92]. Another potential approach is to co-embed datasets in a common space first via a more efficient but potentially less accurate algorithm first, for example using coupled autoencoders, and then perform matching in this space via optimal transport that directly compares samples in the embedding space (rather than employing Gromov-Wasserstein).

The second avenue for improvements is the self-tuning procedure. The heuristics we develop perform significantly better than the default settings of the existing algorithms, which do not have similar self-tuning procedures. This makes the integration task possible even when users do not have validating data, such as cell-type annotations, which are difficult to infer from certain sequencing experiments like HiC (measuring 3D structure of the chromatin). However, the self-tuning heuristics we developed yield a small decrease in alignment performance compared to the hyperparameter tuning case with validation data. Investigating better performing self-tuning methods could make these algorithms more useful in real-world applications.

Lastly, the algorithm presented in the last chapter, SCOOTR, takes the integration task one step forward by also generating hypotheses between feature alignments, without relying on paired data. Algorithms that yield interpretable results on multi-omic relationships can lead to new biological insights by helping scientists prioritize validation experiments. While SCOOTR experiments show encouraging results on multi-omic relationships inferred, future work could improve on these results by (1) introducing a more sophisticated formulation in the feature alignment cost computation step of SCOOTR to account for more complex relationships and (2) leveraging

existing approaches used for reducing noise in single-cell datasets, as as pseudobulk analyses. Regarding the first direction here, neural formulations of optimal transport, which can take into more complex relationships between the samples or features than optimal transport alone, have previously yielded impressive results on machine learning benchmarks. However, employing such approaches is not straightforward for the single-cell feature alignment tasks, as these tend to perform alignment in the latent space of the neural models, and not in the original feature space, losing interpretability. Regarding the second direction, optimal transport results will be highly influenced by the quality of the ground cost defined between the features. Since single-cell datasets are notoriously noisy, approaches like pseudobulk analyses could reduce the level of noise in the ground cost. Such an approach could be implemented by grouping cells after the sample alignment step, and taking group averages for the genomic features before computing the cost matrix. This, however, would increase the runtime complexity of the algorithm and would work only if the clusters are biologically meaningful. Finally, due to the size of the feature space in single-cell datasets, we needed to perform feature pruning before running SCOOTR, only keeping features that show high variability across cells. A formulation that jointly picks features when computing alignments could further increase the quality of feature relationships predicted, which is another potential avenue for future work.

Chapter 7

Appendix

7.1 Proofs for SCOOTR

Notations We first define our notations=:

- $\mathbf{X}$, $\mathbf{Y}$: two datasets with n and n' samples in each, from $\mathbf{X} \in \mathbb{R}^d$ and $\mathbf{Y} \in \mathbb{R}^{d'}$, respectively.
- $\hat{\mu}$, $\hat{\nu}$: empirical distributions associated with the samples (rows) of $\mathbf{X}$ and $\mathbf{Y}$, respectively.
- $\hat{\mu'}$, $\hat{\nu'}$: empirical distributions associated with the features (columns) of $\mathbf{X}$ and $\mathbf{Y}$.
- $\mathbf{K_X} \in \mathbb{R}^{n \times n}$, $\mathbf{K_Y} \in \mathbb{R}^{n' \times n'}$: pairwise sample similarity matrices for $\mathbf{X}$ and $\mathbf{Y}$.
- γ^s, γ^v: coupling matrices between samples and features, respectively.

We additionally define:

- $E^{GW}(\mathbf{K_X}, \mathbf{K_Y}, \gamma^s) = \sum_{ijkl} L(\mathbf{K_X}_{ik}, \mathbf{K_Y}_{jl}) \gamma^s_{ij} \gamma^s_{kl}$
- $E^{COOT}(\mathbf{X}, \mathbf{Y}, \gamma^s, \gamma^v) = \sum_{i,j} M_{ij}(\mathbf{X}, \mathbf{Y}, \gamma^v) \gamma^s_{ij}$,
  with $M(\mathbf{X}, \mathbf{Y}, \gamma^v) = \sum_{ij} L(\mathbf{X}_i, \mathbf{Y}_j) \gamma^v$

•

$$E^{FGCOOT}(\mathbf{X}, \mathbf{Y}, \mathbf{K_X}, \mathbf{K_Y}, \gamma^s, \gamma^v) = \alpha \sum_{ijkl} L(\mathbf{K_X}_{ik}, \mathbf{K_Y}_{jl}) \gamma^s_{ij} \gamma^s_{kl} + (1-\alpha) \sum_{ij} M_{ij}(\mathbf{X}, \mathbf{Y}, \gamma^v) \gamma^s_{ij}$$

such that:

$$\begin{aligned} GW(\hat{\mu}, \hat{\nu}) &= \min_{\gamma^s \in \Gamma(\hat{\mu}, \hat{\nu})} E^{GW}(\mathbf{K_X}, \mathbf{K_Y}, \gamma^s) \\ COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) &= \min_{\substack{\gamma^s \in \Gamma(\hat{\mu}, \hat{\nu}), \\ \gamma^v \in \Gamma(\hat{\mu'}, \hat{\nu'})}} E^{COOT}(\mathbf{X}, \mathbf{Y}, \gamma^s, \gamma^v) \\ FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) &= \min_{\substack{\gamma^s \in \Gamma(\hat{\mu}, \hat{\nu}), \\ \gamma^v \in \Gamma(\hat{\mu'}, \hat{\nu'})}} E^{FGCOOT}(\mathbf{X}, \mathbf{Y}, \mathbf{K_X}, \mathbf{K_Y}, \gamma^s, \gamma^v) \end{aligned}$$

Bounds

Lemma 1: $FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'})$ is lower-bounded by the interpolation between $GW(\hat{\mu}, \hat{\nu})$ and $COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'})$:

$$FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) \geq \alpha GW(\hat{\mu}, \hat{\nu}) + (1-\alpha) COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) \tag{7.1}$$

Proof. Let $\alpha^{s,\alpha}$ be the coupling that minimizes E^{FGCOOT}. Then,

$$FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) := E^{FGCOOT}(\mathbf{X}, \mathbf{Y}, \mathbf{K_X}, \mathbf{K_Y}, \gamma^{s,\alpha}, \gamma^v) \tag{7.2}$$

Since $\gamma^{s,\alpha}$ may not necessarily be the optimal coupling for $COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'})$ or for $GW(\hat{\mu}, \hat{\nu})$, we have:

$$COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) \leq E^{COOT}(\mathbf{X}, \mathbf{Y}, \gamma^{s,\alpha}, \gamma^v) \tag{7.3}$$

$$COOT(\hat{\mu}, \hat{\nu}, \hat{\mu'}, \hat{\nu'}) GW(\hat{\mu}, \hat{\nu}) \leq E^{GW}(\mathbf{K_X}, \mathbf{K_Y}, \gamma^{s,\alpha}) \tag{7.4}$$

The inequality in Equation 7.1 is then derived. □

Equations 7.3 and 7.4 also imply:

$$FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') \geq (1-\alpha)COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') \tag{7.5}$$

$$FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') \geq \alpha GW(\hat{\mu}, \hat{\nu}) \tag{7.6}$$

Interpolation Properties

Theorem 1: *As α tends to zero, the $FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$ recovers $COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$ and as α tends to 1, it recovers $GW(\hat{\mu}, \hat{\nu})$:*

$$\lim_{\alpha \to 0} FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') = COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$$

$$\lim_{\alpha \to 1} FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') = GW(\hat{\mu}, \hat{\nu})$$

Proof. We declare that

- $\gamma^{s,\alpha}$ denotes the optimal sample coupling for $FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$,
- $\gamma^{s,COOT}$ denotes the optimal sample coupling for $COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$,
- $\gamma^{s,GW}$ denotes the optimal sample coupling for $GW(\hat{\mu}, \hat{\nu})$.

Then, due to the suboptimality of $\gamma^{s,\alpha}$ for $COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$ and $\gamma^{s,COOT}$ for $FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')$ we have:

$$\underbrace{FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') - (1-\alpha)COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}')}_{E^{FGCOOT}(*, \gamma^{s,\alpha}, \gamma^v) - (1-\alpha)E^{COOT}(*, \gamma^{s,COOT}, \gamma^v)} \tag{7.7}$$

$$\leq \underbrace{E^{FGCOOT}(*, \gamma^{s,COOT}, \gamma^v) - (1-\alpha)E^{COOT}(*, \gamma^{s,COOT}, \gamma^v)}_{\alpha E^{GW}(*, \gamma^{s,COOT})} \tag{7.8}$$

$$\implies FGCOOT_\alpha(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') \leq (1-\alpha)COOT(\hat{\mu}, \hat{\nu}, \hat{\mu}', \hat{\nu}') + \alpha E^{GW}(*, \gamma^{s,COOT}) \tag{7.9}$$

Taking Equation7.9(14) and Equation7.5 together:

$$(1-\alpha)COOT(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') \leq FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') \tag{7.10}$$

$$\leq (1-\alpha)COOT(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') + \alpha E^{GW}(*,\gamma^{s,COOT}) \tag{7.11}$$

Therefore, as α goes to 0, $\lim_{\alpha\to 0} FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') = COOT(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}')$.

Similarly, we can use the suboptimality of of $\gamma^{s,\alpha}$ for $GW(\hat{\mu},\hat{\nu})$ and $\gamma^{s,GW}$ for $FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}')$:

$$\underbrace{FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') - \alpha GW(\hat{\mu},\hat{\nu})}_{E^{FGCOOT}(*,\gamma^{s,\alpha},\gamma^v)-\alpha E^{GW}(*,\gamma^{s,GW})} \leq \underbrace{E^{FGCOOT}(*,\gamma^{s,GW},\gamma^v) - \alpha E^{GW}(*,\gamma^{s,GW})}_{(1-\alpha)E^{COOT}(*,\gamma^{s,COOT},\gamma^v)} \tag{7.12}$$

$$\implies FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') \leq \alpha GW(\hat{\mu},\hat{\nu}) + \alpha E^{GW}(*,\gamma^{s,GW},\gamma^v) \tag{7.13}$$

Taking Equations 7.6 and 7.13 together:

$$\alpha GW(\hat{\mu},\hat{\nu}) \leq FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') \leq \alpha GW(\hat{\mu},\hat{\nu}) + \alpha E^{COOT}(*,\gamma^{s,GW}) \tag{7.14}$$

Therefore, as α goes to 1, $\lim_{\alpha\to 1} FGCOOT_\alpha(\hat{\mu},\hat{\nu},\hat{\mu}',\hat{\nu}') = GW(\hat{\mu},\hat{\nu})$. □

7.2 Python implementation of the proposed algorithms

For full implementation of the proposed algorithms, along with example applications that reproduce results in this thesis ad utility functions that preprocess data etc, please visit the following GitHub repositories:

- **For SCOT and SCOTv2**: https://github.com/rsinghlab/SCOT
- **For SCOOTR**: https://github.com/rsinghlab/SCOOTR

However, the code included below present the crux of these algorithms, implemented in Python:

Code for the SCOT algorithm

```
"""
Authors: Pinar Demetci, Rebecca Santorella
Principal Investigator: Ritambhara Singh, Ph.D. from Brown University
12 February 2020
Updated: 27 November 2020
SCOT algorithm (version 1): Single Cell alignment using Optimal Transport
Correspondence: pinar_demetci@brown.edu, rebecca_santorella@brown.edu,
                                        ritambhara@brown.edu
"""

### Import python packages we depend on:
# For regular matrix operations:
import numpy as np
# For optimal transport operations:
import ot
# For computing graph distances:
from scipy.sparse.csgraph import dijkstra
from scipy.sparse import csr_matrix
from sklearn.neighbors import kneighbors_graph

# For pre-processing, normalization
from sklearn.preprocessing import StandardScaler, normalize


class SCOT(object):
```

```
"""
SCOT algorithm for unsupervised alignment of single-cell multi-omic data.
https://www.biorxiv.org/content/10.1101/2020.04.28.066787v2 (original
                                    preprint)
https://www.liebertpub.com/doi/full/10.1089/cmb.2021.0446 (Journal of
                                    Computational Biology publication
                                    through RECOMB 2021 conference)
Input: domain1, domain2 in form of numpy arrays/matrices, where the rows
                                    correspond to samples and columns
                                    correspond to features.
Returns: aligned domain 1, aligned domain 2 in form of numpy arrays/
                                    matrices projected on domain 1
Example use:
# Given two numpy matrices, domain1 and domain2, where the rows are cells
                                    and columns are different genomic
                                    features:
scot= SCOT(domain1, domain2)
aligned_domain1, aligned_domain2 = scot.align(k=20, e=1e-3)
#If you can't pick the parameters k and e, you can try out our
                                    unsupervised self-tuning heuristic by
                                    running:
scot= SCOT(domain1, domain2)
aligned_domain1, aligned_domain2 = scot.align(selfTune=True)
Required parameters:
- k: Number of neighbors to be used when constructing kNN graphs. Default=
                                     min(min(n_1, n_2), 50), where n_i,
                                    for i=1,2 corresponds to the number of
                                     samples in the i^th domain.
- e: Regularization constant for the entropic regularization term in
                                    entropic Gromov-Wasserstein optimal
                                    transport formulation. Default= 1e-3

Optional parameters:
- normalize= Determines whether to normalize input data ahead of alignment
                                    . True or False (boolean parameter).
                                    Default = True.
- norm= Determines what sort of normalization to run, "l2", "l1", "max", "
                                    zscore". Default="l2"
```

```
- mode: "connectivity" or "distance". Determines whether to use a
                                    connectivity graph (adjacency matrix
                                    of 1s/0s based on whether nodes are
                                    connected) or a distance graph (
                                    adjacency matrix entries weighted by
                                    distances between nodes). Default="
                                    connectivity"
- metric: Sets the metric to use while constructing nearest neighbor
                                    graphs. some possible choices are "
                                    correlation", "minkowski".  "
                                    correlation" is Pearson's correlation
                                    and "minkowski" is equivalent to
                                    Euclidean distance in its default form
                                     (). Default= "correlation".
- verbose: Prints loss while optimizing the optimal transport formulation.
                                     Default=True
- XontoY: Determines the direction of barycentric projection. True or
                                    False (boolean parameter). If True,
                                    projects domain1 onto domain2. If
                                    False, projects domain2 onto domain1.
                                    Default=True.
Note: If you want to specify the marginal distributions of the input
                                    domains and not use uniform
                                    distribution, please set the
                                    attributes p and q to the
                                    distributions of your choice (for
                                    domain 1, and 2, respectively)
    after initializing a SCOT class instance and before running alignment
                                    and set init_marginals=False in .align
                                    () parameters
"""

def __init__(self, domain1, domain2):

  self.X=domain1
  self.y=domain2

  self.p= None #empirical probability distribution for domain 1 (X)
  self.q= None #empirical probability distribution for domain 2 (y)

  self.Cx=None #intra-domain graph distances for domain 1 (X)
  self.Cy=None #intra-domain graph distances for domain 2 (y)
```

```
    self.coupling=None # Coupling matrix that relates domain 1 and domain 2,
                                        ..., m
    self.gwdist=None # Gromov-Wasserstein distance between domains after
                                        alignment
    self.flag = None # convergence flag

    self.X_aligned=None #aligned datasets to return: domain1
    self.y_aligned=None #aligned datasets to return: domain2

  def init_marginals(self):
    # Without any prior information, we set the probabilities to what we
                                        observe empirically: uniform over all
                                        observed sample
    self.p= ot.unif(self.X.shape[0])
    self.q = ot.unif(self.y.shape[0])

  def normalize(self, norm="l2", bySample=True):
    assert (norm in ["l1","l2","max", "zscore"]), "Norm argument has to be
                                        either one of 'max', 'l1', 'l2' or '
                                        zscore'. If you would like to perform
                                        another type of normalization, please
                                        give SCOT the normalize data and set
                                        the argument normalize=False when
                                        running the algorithm."

    if (bySample==True or bySample==None):
      axis=1
    else:
      axis=0

    if norm=="zscore":
      scaler=StandardScaler()
      self.X, self.y=scaler.fit_transform(self.X), scaler.fit_transform(self
                                        .y)

    else:
      self.X, self.y =normalize(self.X, norm=norm, axis=axis), normalize(
                                        self.y, norm=norm, axis=axis)

  def construct_graph(self, k, mode= "connectivity", metric="correlation"):
    assert (mode in ["connectivity", "distance"]), "Norm argument has to be
                                        either one of 'connectivity', or '
                                        distance'. "
```

```
    if mode=="connectivity":
      include_self=True
    else:
      include_self=False

    self.Xgraph=kneighbors_graph(self.X, k, mode=mode, metric=metric,
                                 include_self=include_self)
    self.ygraph=kneighbors_graph(self.y, k, mode=mode, metric=metric,
                                 include_self=include_self)

    return self.Xgraph, self.ygraph

  def init_distances(self):
    # Compute shortest distances
    X_shortestPath=dijkstra(csgraph= csr_matrix(self.Xgraph), directed=False
                                 , return_predecessors=False)
    y_shortestPath=dijkstra(csgraph= csr_matrix(self.ygraph), directed=False
                                 , return_predecessors=False)

    # Deal with unconnected stuff (infinities):
    X_max=np.nanmax(X_shortestPath[X_shortestPath != np.inf])
    y_max=np.nanmax(y_shortestPath[y_shortestPath != np.inf])
    X_shortestPath[X_shortestPath > X_max] = X_max
    y_shortestPath[y_shortestPath > y_max] = y_max

    # Finnally, normalize the distance matrix:
    self.Cx=X_shortestPath/X_shortestPath.max()
    self.Cy=y_shortestPath/y_shortestPath.max()

    return self.Cx, self.Cy

  def find_correspondences(self, e, verbose=True):
    self.coupling, log= ot.gromov.entropic_gromov_wasserstein(self.Cx, self.
                                 Cy, self.p, self.q, loss_fun='
                                 square_loss', epsilon=e, log=True,
                                 verbose=verbose)
    self.gwdist=log['gw_dist']

    # Check convergence:
    if (np.isnan(self.coupling).any() or np.any(~self.coupling.any(axis=1))
                                 or np.any(~self.coupling.any(axis=0))
                                 or sum(sum(self.coupling)) < .95):
      self.flag=False
```

```
    else:
      self.flag=True

    return self.gwdist

  def barycentric_projection(self, XontoY=True):
    if XontoY:
      #Projecting the first domain onto the second domain
      self.y_aligned=self.y
      weights=np.sum(self.coupling, axis = 0)
      self.X_aligned=np.matmul(self.coupling, self.y) / weights[:, None]
    else:
      #Projecting the second domain onto the first domain
      self.X_aligned=self.X
      weights=np.sum(self.coupling, axis = 0)
      self.y_aligned=np.matmul(np.transpose(self.coupling), self.X) /
                                        weights[:, None]
    return self.X_aligned, self.y_aligned

  def align(self, k=None, e=1e-3, mode="connectivity", metric="correlation",
                                         verbose=True, normalize=True, norm="
                                        l2", XontoY=True, selfTune=False,
                                        init_marginals=True):
    if normalize:
      self.normalize(norm=norm)
    if init_marginals:
      self.init_marginals()

    if selfTune:
      X_aligned, y_aligned= self.unsupervised_scot()
    else:
      if k==None:
        k=min((int(self.X.shape[0]*0.2), int(self.y.shape[0]*0.2)),50)

      self.construct_graph(k, mode= "connectivity", metric="correlation")
      self.init_distances()
      self.find_correspondences(e=e, verbose=verbose)

      if self.flag==False:
        print("CONVERGENCE ERROR: Optimization procedure runs into numerical
                                         errors with the hyperparameters
                                        specified. Please try aligning with
                                        higher values of epsilon.")
```

```
        return

      else:
        X_aligned, y_aligned = self.barycentric_projection(XontoY=XontoY)

    self.X_aligned, self.y_aligned=X_aligned, y_aligned
    return self.X_aligned, self.y_aligned

  def search_scot(self, ks, es, all_values = False,  mode= "connectivity",
                                      metric="correlation", normalize=True,
                                      norm="l2", init_marginals=True):
    '''
    Performs a hyperparameter sweep for given values of k and epsilon
    Default: return the parameters corresponding to the lowest GW distance
    (Optional): return all k, epsilon, and GW values
    '''

    # initialize alignment
    if normalize:
      self.normalize(norm=norm)
    if init_marginals:
      self.init_marginals()

    # Note to self: Incorporate multiprocessing here to speed things up
    # store values of k, epsilon, and gw distance
    total=len(es)*len(ks)
    k_sweep=np.zeros(total)
    e_sweep=np.zeros(total)
    gw_sweep=np.zeros(total)

    gmin = 1
    counter=1

    X_aligned,y_aligned=None, None
    e_best,k_best=None, None
    # search in k first to reduce graph computation
    for k in ks:
      self.construct_graph(k, mode= mode, metric=metric)
      self.init_distances()
      for e in es:
        print(counter, "/", total)
        print("Aligning k: ",k, " and e: ",e)
        # run alignment / optimize correspondence matrix:
```

```
        self.find_correspondences(e=e, verbose=False)
        # save values
        if self.flag:
          if all_values:
            k_sweep[counter]=k
            e_sweep[counter]=e
            gw_sweep[counter] = self.gwdist

            print(self.gwdist)
          # save the alignment if it is lower
          if self.gwdist < gmin:
            X_aligned, y_aligned = self.barycentric_projection()
            gmin =self.gwdist
            e_best, k_best= e, k
          counter = counter + 1

    if all_values:
      # return alignment and all values
      return X_aligned, y_aligned, gw_sweep, k_sweep, e_sweep
    else:
      # return  alignment and the parameters corresponding to the lowest GW
                                          distance
      return X_aligned, y_aligned, gmin, k_best, e_best


def unsupervised_scot(self, normalize=False, norm='l2'):
    '''
    Unsupervised hyperparameter tuning algorithm to find an alignment
    by using the GW distance as a measure of alignment
    '''

    # use k = 20% of # sample or k = 50 if dataset is large
    n = min(self.X.shape[0], self.y.shape[0])
    k_start = min(n // 5, 50)

    num_eps = 12
    num_k = 5

    # define search space
    es = np.logspace(-1, -3, num_eps)
    if ( n > 250):
      ks = np.linspace(20, 100, num_k)
    else:
```

```
        ks = np.linspace(n//20, n//6, num_k)
    ks = ks.astype(int)

    # search parameter space
    X_aligned, y_aligned, g_best, k_best, e_best = self.search_scot(ks, es,
                                        all_values=False, normalize=normalize,
                                         norm=norm, init_marginals=False)

    print("Alignment completed. Hyperparameters selected from the
                                        unsupervised hyperparameter sweep are:
                                         %d for number of neighbors k and %f
                                        for epsilon" %(k_best, e_best))

    return X_aligned,
```

Code for the SCOTv2 algorithm

```
    """
Author: Pinar Demetci
Principal Investigator: Ritambhara Singh, Ph.D. from Brown University
08 August 2021
Updated: 23 February 2023
SCOTv2 algorithm: Single Cell alignment using Optimal Transport version 2
Correspondence: pinar_demetci@brown.edu, ritambhara@brown.edu
"""

### Import python packages we depend on:
import numpy as np
import torch
import ot
import scipy
# For computing graph distances:
from scipy.sparse.csgraph import dijkstra
from scipy.sparse import csr_matrix
from sklearn.neighbors import kneighbors_graph

# For pre-processing, normalization
from sklearn.preprocessing import StandardScaler, normalize


class SCOTv2(object):
  """
  SCOT algorithm for unsupervised alignment of single-cell multi-omic data.
  https://www.biorxiv.org/content/10.1101/2020.04.28.066787v2 (original
                                    preprint)
  https://www.liebertpub.com/doi/full/10.1089/cmb.2021.0446 (Journal of
                                    Computational Biology publication
                                    through RECOMB 2021 conference)

  Input: domain1, domain2 in form of numpy arrays/matrices, where the rows
                                    correspond to samples and columns
                                    correspond to features.
  Returns: aligned domain 1, aligned domain 2 in form of numpy arrays/
                                    matrices projected on domain 1

  Example use:
```

```
# Given two numpy matrices, domain1 and domain2, where the rows are cells
                                    and columns are different genomic
                                    features:
scot= SCOT(domain1, domain2)
aligned_domain1, aligned_domain2 = scot.align(k=20, e=1e-3)

#If you can't pick the parameters k and e, you can try out our
                                    unsupervised self-tuning heuristic by
                                    running:
scot= SCOT(domain1, domain2)
aligned_domain1, aligned_domain2 = scot.align(selfTune=True)

Required parameters:
- k: Number of neighbors to be used when constructing kNN graphs. Default=
                                     min(min(n_1, n_2), 50), where n_i,
                                    for i=1,2 corresponds to the number of
                                     samples in the i^th domain.
- e: Regularization constant for the entropic regularization term in
                                    entropic Gromov-Wasserstein optimal
                                    transport formulation. Default= 1e-3

Optional parameters:

- normalize= Determines whether to normalize input data ahead of alignment
                                    . True or False (boolean parameter).
                                    Default = True.
- norm= Determines what sort of normalization to run, "l2", "l1", "max", "
                                    zscore". Default="l2"
- mode: "connectivity" or "distance". Determines whether to use a
                                    connectivity graph (adjacency matrix
                                    of 1s/0s based on whether nodes are
                                    connected) or a distance graph (
                                    adjacency matrix entries weighted by
                                    distances between nodes). Default="
                                    connectivity"
- metric: Sets the metric to use while constructing nearest neighbor
                                    graphs. some possible choices are "
                                    correlation", "minkowski".  "
                                    correlation" is Pearson's correlation
                                    and "minkowski" is equivalent to
                                    Euclidean distance in its default form
                                     (). Default= "correlation".
```

```
- verbose: Prints loss while optimizing the optimal transport formulation.
                                    Default=True
- XontoY: Determines the direction of barycentric projection. True or
                                    False (boolean parameter). If True,
                                    projects domain1 onto domain2. If
                                    False, projects domain2 onto domain1.
                                    Default=True.

Note: If you want to specify the marginal distributions of the input
                                    domains and not use uniform
                                    distribution, please set the
                                    attributes p and q to the
                                    distributions of your choice (for
                                    domain 1, and 2, respectively)
    after initializing a SCOT class instance and before running alignment
                                    and set init_marginals=False in .align
                                    () parameters
"""

def __init__(self, data):

  assert type(data)==list and len(data)>=2, "As input, SCOTv2 requires a
                                    list, containing at least two numpy
                                    arrays to be aligned.  \
      Each numpy array/matrix corresponds to a dataset, with samples (
                                    cells) in rows and features (latent
                                    representations or genomic features)
                                    in columns. \
      We recommend using latent representations (e.g. principal components
                                     for RNA-seq and topics - via cisTopic
                                    - for ATAC-seq/Methyl-seq)."
  self.data=data
  self.marginals=[] # Holds the empirical probability distributions over
                                    samples in each dataset
  self.graphs=[] # Holds graphs per dataset
  self.graphDists=[] # Holds intra-domain graph distances for each input
                                    dataset
  self.couplings=[] # Holds coupling matrices
  self.gwdists=[] # Gromov-Wasserstein distances between domains after
                                    alignment
  self.flags = [] # Holds alignment convergence flags (booleans: True/
                                    False)
```

```
    self.aligned_data=[]

  def _init_marginals(self):
    # Without any prior information, we set the probabilities to what we
                                        observe empirically: uniform over all
                                        observed sample
    for i in range(len(self.data)):
      num_cells=self.data[i].shape[0]
      marginalDist=torch.ones(num_cells)/num_cells
      self.marginals.append(marginalDist)
    return self.marginals

  def _normalize(self, norm="l2", bySample=True):
    assert (norm in ["l1","l2","max", "zscore"]), "Norm argument has to be
                                        either one of 'max', 'l1', 'l2' or '
                                        zscore'.\
     If you would like to perform another type of normalization, please give
                                         SCOT the normalized data and set the
                                        argument 'normalize=False' when
                                        running the algorithm. \
     We have found l2 normalization to empirically perform better with
                                        single-cell sequencing datasets,
                                        including when using latent
                                        representations. "

    for i in range(len(self.data)):
      if norm=="zscore":
        scaler=StandardScaler()
        self.data[i]=scaler.fit_transform(self.data[i])
      else:
        if (bySample==True or bySample==None):
          axis=1
        else:
          axis=0
        self.data[i] =normalize(self.data[i], norm=norm, axis=axis)
    return self.data # Normalized data

  def construct_graph(self, k=20, mode= "connectivity", metric="correlation"
                                        ):
    assert (mode in ["connectivity", "distance"]), "Norm argument has to be
                                        either one of 'connectivity', or '
                                        distance'. "
    if mode=="connectivity":
```

```
    include_self=True
  else:
    include_self=False

  for i in range(len(self.data)):
    self.graphs.append(kneighbors_graph(self.data[i], n_neighbors=k, mode=
                                        mode, metric=metric, include_self=
                                        include_self))

  return self.graphs

def init_graph_distances(self):
  for i in range(len(self.data)):
    # Compute shortest distances
    shortestPath=dijkstra(csgraph= csr_matrix(self.graphs[i]), directed=
                                        False, return_predecessors=False)
    # Deal with unconnected stuff (infinities):
    Max_dist=np.nanmax(shortestPath[shortestPath != np.inf])
    shortestPath[shortestPath > Max_dist] = Max_dist
    # Finnally, normalize the distance matrix:
    self.graphDists.append(shortestPath/shortestPath.max())

  return self.graphDists

def _exp_sinkhorn_solver(self, ecost, u, v,a,b, mass, eps, rho, rho2,
                                        nits_sinkhorn, tol_sinkhorn):
    """
    Parameters
    ----------
    - ecost: torch.Tensor of size [size_X, size_Y]
         Exponential kernel generated from the local cost based on the
                                        current coupling.
    - u: torch.Tensor of size [size_X[0]].
       First dual potential defined on X.
    - v: torch.Tensor of size [size_Y[0]].
       Second dual potential defined on Y.
    - mass: torch.Tensor of size [1].
        Mass of the current coupling.
    - nits_sinkhorn: int.
            Maximum number of iterations to update Sinkhorn potentials in
                                        inner loop.
    - tol_sinkhorn: float
           Tolerance on convergence of Sinkhorn potentials.
```

```
    Returns
    ----------
    u: torch.Tensor of size [size_X[0]]
       First dual potential of Sinkhorn algorithm
    v: torch.Tensor of size [size_Y[0]]
       Second dual potential of Sinkhorn algorithm
    logpi: torch.Tensor of size [size_X, size_Y]
         Optimal transport plan in log-space.
    """
    # Initialize potentials by finding best translation
    if u is None or v is None:
      u, v = torch.ones_like(a), torch.ones_like(b)
    k = (a * u ** (-eps / rho)).sum()+ (b * v ** (-eps / rho)).sum()
    k = k / (2 * (u[:, None] * v[None, :] * ecost *a[:, None] * b[None, :]
                                        ).sum())
    z = (0.5 * mass * eps) / (2.0 + 0.5 * (eps / rho) + 0.5 * (eps / rho2)
                                        )
    k = k ** z
    u,v= u * k, v * k

    # perform Sinkhorn updates in LSE form
    for j in range(nits_sinkhorn):
      u_prev = u.clone()
      v = torch.einsum("ij,i->j", ecost, a * u) ** (-1.0 / (1.0 + eps /
                                        rho))
      u = torch.einsum("ij,j->i", ecost, b * v) ** (-1.0 / (1.0 + eps /
                                        rho2))
      if (u.log() - u_prev.log()).abs().max().item() * eps < tol_sinkhorn:
        break
    pi = u[:, None] * v[None, :] * ecost * a[:, None] * b[None, :]
    return u, v, pi

def exp_unbalanced_gw(self,a, dx, b, dy, eps=0.01, rho=1.0, rho2=None,
                                        nits_plan=3000, tol_plan=1e-6,
                                        nits_sinkhorn=3000, tol_sinkhorn=1e-6)
                                        :
  if rho2 is None:
    rho2 = rho #KL divergence coefficient doesn't have to be the same for
                                        both couplings.
          #But, to keep #hyperparameters low, we default to using the
                                        same coefficient.
```

```
        #Someone else playing with our code could assign a rho2
                                    different than rho, though.

# Initialize the coupling and local costs
pi= a[:, None]* b[None, :] / (a.sum() * b.sum()).sqrt()
pi_prev = torch.zeros_like(pi)
up, vp = None, None

for i in range(nits_plan):
  pi_prev = pi.clone()
  mp = pi.sum()

  #Compute the current local cost:
  distxy = torch.einsum("ij,kj->ik", dx, torch.einsum("kl,jl->kj", dy,
                                    pi))
  kl_pi = torch.sum(pi * (pi / (a[:, None] * b[None, :]) + 1e-10).log())
  mu, nu = torch.sum(pi, dim=1), torch.sum(pi, dim=0)
  distxx = torch.einsum("ij,j->i", dx ** 2, mu)
  distyy = torch.einsum("kl,l->k", dy ** 2, nu)
  lcost = (distxx[:, None] + distyy[None, :] - 2 * distxy) + eps * kl_pi
  if rho < float("Inf"):
    lcost = (lcost+ rho* torch.sum(mu * (mu / a + 1e-10).log()))
  if rho2 < float("Inf"):
    lcost = (lcost+ rho2* torch.sum(nu * (nu / b + 1e-10).log()))
  ecost = (-lcost /(mp * eps)).exp()

  if (i%10)==0:
    print("Unbalanced GW step:", i)
  #compute the coupling via sinkhorn
  up, vp, pi = self._exp_sinkhorn_solver(ecost, up, vp, a, b, mp, eps,
                                    rho, rho2,nits_sinkhorn, tol_sinkhorn)

  flag=True
  if torch.any(torch.isnan(pi)):
    flag=False

  pi = (mp / pi.sum()).sqrt() * pi
  if (pi - pi_prev).abs().max().item() < tol_plan:
    break
return pi, flag
```

```
def find_correspondences(self, normalize=True, norm="l2", bySample=True, k
                                    =20, mode= "connectivity", metric="
                                    correlation",  eps=0.01, rho=1.0, rho2
                                    =None):
  # Normalize
  if normalize:
    self._normalize(norm=norm, bySample=bySample)
  # Initialize inputs for (unbalanced) Gromov-Wasserstein optimal
                                    transport:
  self._init_marginals()
  print("computing intra-domain graph distances")
  self.construct_graph(k=k, mode=mode, metric=metric)
  self.init_graph_distances()
  # Run pairwise dataset alignments:
  for i in range(len(self.data)-1):
    print("running pairwise dataset alignments")
    a,b =torch.Tensor(self.marginals[0]), torch.Tensor(self.marginals[i+1]
                                    )
    dx, dy= torch.Tensor(self.graphDists[0]), torch.Tensor(self.graphDists
                                    [i+1])
    coupling, flag=self.exp_unbalanced_gw(a, dx, b, dy, eps=eps, rho=rho,
                                    rho2=rho2, nits_plan=3000, tol_plan=1e
                                    -6, nits_sinkhorn=3000, tol_sinkhorn=
                                    1e-6)
    self.couplings.append(coupling)
    self.flags.append(flag)
    if flag==False:
        raise Exception(
        f"Solver got NaN plan with params (eps, rho, rho2) "
        f" = {eps, rho, rho2}. Try increasing argument eps")
  return self.couplings

def barycentric_projection(self):
  aligned_datasets=[self.data[0]]
  for i in range(0,len(self.couplings)):
    coupling=np.transpose(self.couplings[i].numpy())
    weights=np.sum(coupling, axis = 1)
    projected_data=np.matmul((coupling/ weights[:, None]), self.data[0])
    aligned_datasets.append(projected_data)
  return aligned_datasets

def coembed_datasets(self, Lambda=1.0, out_dim=10):
  """
```

```
    Co-embeds datasets in a shared space.
    Implementation is based on Cao et al 2022 (Pamona)
    """
    n_datasets = len(self.data)
    H0 = []
    L = []
    for i in range(n_datasets-1):
      self.couplings[i] = self.couplings[i]*np.shape(self.data[i])[0]

    for i in range(n_datasets):
      graph_data = self.graphs[i] + self.graphs[i].T.multiply(self.graphs[i]
                                        .T > self.graphs[i]) - \
        self.graphs[i].multiply(self.graphs[i].T > self.graphs[i])
      W = np.array(graph_data.todense())
      index_pos = np.where(W>0)
      W[index_pos] = 1/W[index_pos]
      D = np.diag(np.dot(W, np.ones(np.shape(W)[1])))
      L.append(D - W)

    Sigma_x = []
    Sigma_y = []
    for i in range(n_datasets-1):
      Sigma_y.append(np.diag(np.dot(np.transpose(np.ones(np.shape(self.
                                        coupling[i])[0])), self.coupling[i])))
      Sigma_x.append(np.diag(np.dot(self.coupling[i], np.ones(np.shape(self.
                                        coupling[i])[1]))))

    S_xy = coupling[0]
    S_xx = L[0] + Lambda*Sigma_x[0]
    S_yy = L[-1] +Lambda*Sigma_y[0]
    for i in range(1, n_datasets-1):
      S_xy = np.vstack((S_xy, self.coupling[i]))
      S_xx = block_diag(S_xx, L[i] + Lambda*Sigma_x[i])
      S_yy = S_yy + Lambda*Sigma_y[i]

    v, Q = np.linalg.eig(S_xx)
    v = v + 1e-12
    V = np.diag(v**(-0.5))
    H_x = np.dot(Q, np.dot(V, np.transpose(Q)))

    v, Q = np.linalg.eig(S_yy)
    v = v + 1e-12
    V = np.diag(v**(-0.5))
```

```
    H_y = np.dot(Q, np.dot(V, np.transpose(Q)))

    H = np.dot(H_x, np.dot(S_xy, H_y))
    U, sigma, V = np.linalg.svd(H)

    num = [0]
    for i in range(n_datasets-1):
      num.append(num[i]+len(data[i]))

    U, V = U[:,:output_dim], np.transpose(V)[:,:output_dim]

    fx = np.dot(H_x, U)
    fy = np.dot(H_y, V)

    integrated_data = []
    for i in range(n_datasets-1):
      integrated_data.append(fx[num[i]:num[i+1]])

    integrated_data.append(fy)

    return integrated_data

def align(self,normalize=True, norm="l2", bySample=True, k=20, mode= "
                                    connectivity", metric="correlation",
                                    eps=0.01, rho=1.0, rho2=None,
                                    projMethod="embedding", Lambda=1.0,
                                    out_dim=10):
    assert projMethod in ["embedding", "barycentric"], "The input to the
                                    parameter 'projMethod' needs to be one
                                     of \
                'embedding' (if co-embedding them in a new shared space) or
                                    'barycentric' (if using barycentric
                                    projection)"
    self.find_correspondences(normalize=normalize, norm=norm, bySample=
                                    bySample, k=k, mode=mode, metric=
                                    metric,  eps=eps, rho=rho, rho2=rho2)
    print("FLAGS", self.flags)
    if projMethod=="embedding":
      integrated_data=self.coembed_datasets(Lambda=Lambda, out_dim=out_dim)
    else:
      integrated_data=self.barycentric_projection()
    self.integrated_data=integrated_data
    return integrated_data
```

Code for the SGCOOTR algorithm

```
    import numpy as np
import ot
from scipy import stats
from scipy.sparse import random
from ot.bregman import sinkhorn_scaling

def random_gamma_init(p,q, **kwargs):
  """ Returns random coupling matrix with marginal p,q
  """
  rvs=stats.beta(1e-1,1e-1).rvs
  S=random(len(p), len(q), density=1, data_rvs=rvs)
  return sinkhorn_scaling(p,q,S.A, **kwargs)

def init_matrix_np(X1, X2, v1, v2):
  """Return loss matrices and tensors for COOT fast computation
  Returns the value of |X1-X2|^{2} \otimes T as done in [1] based on [2] for
                                        the Gromov-Wasserstein distance.
  Where :
    - X1 : The source dataset of shape (n,d)
    - X2 : The target dataset of shape (n',d')
    - v1 ,v2 : weights (histograms) on the columns of resp. X1 and X2
    - T : Coupling matrix of shape (n,n')
  Parameters
  ----------
  X1 : numpy array, shape (n, d)
     Source dataset
  X2 : numpy array, shape (n', d')
     Target dataset
  v1 : numpy array, shape (d,)
    Weight (histogram) on the features of X1.
  v2 : numpy array, shape (d',)
    Weight (histogram) on the features of X2.

  Returns
  -------
  constC : ndarray, shape (n, n')
    Constant C matrix (see paragraph 1.2 of supplementary material in [1])
  hC1 : ndarray, shape (n, d)
    h1(X1) matrix (see paragraph 1.2 of supplementary material in [1])
  hC2 : ndarray, shape (n', d')
    h2(X2) matrix (see paragraph 1.2 of supplementary material in [1])
```

```
  References
  ----------
  .. [1] Redko Ievgen, Vayer Titouan, Flamary R{\'e}mi and Courty Nicolas
      "CO-Optimal Transport"
  .. [2] Peyr , Gabriel, Marco Cuturi, and Justin Solomon,
    "Gromov-Wasserstein averaging of kernel and distance matrices."
    International Conference on Machine Learning (ICML). 2016.
  """
  def f1(a):
    return (a ** 2)

  def f2(b):
    return (b ** 2)

  def h1(a):
    return a

  def h2(b):
    return 2 * b

  constC1 = np.dot(np.dot(f1(X1), v1.reshape(-1, 1)),
           np.ones(f1(X2).shape[0]).reshape(1, -1))
  constC2 = np.dot(np.ones(f1(X1).shape[0]).reshape(-1, 1),
           np.dot(v2.reshape(1, -1), f2(X2).T))

  constC = constC1 + constC2
  hX1 = h1(X1)
  hX2 = h2(X2)

  return constC, hX1, hX2


def fgcot(X1, X2, C1,C2, w1 = None, w2 = None, v1 = None, v2 = None, alpha=0
                                        .5,
         niter=100, algo='sinkhorn', reg=0,algo2='sinkhorn',
         reg2=0, verbose=True, log=False, random_init=False, C_lin=None):

  """ Returns COOT between two datasets X1,X2 (see [1])

  The function solves the following optimization problem:
  .. math::
    COOT = \min_{Ts,Tv} \sum_{i,j,k,l} |X1_{i,k}-X2_{j,l}|^{2}*Ts_{i,j}*Tv_{
                                        k,l}
```

```
Where :
- X1 : The source dataset
- X2 : The target dataset
- w1,w2  : weights (histograms) on the samples (rows) of resp. X1 and X2
- v1,v2  : weights (histograms) on the features (columns) of resp. X1 and
                                    X2

Parameters
----------
X1 : numpy array, shape (n, d)
   Source dataset
X2 : numpy array, shape (n', d')
   Target dataset
w1 : numpy array, shape (n,)
  Weight (histogram) on the samples of X1. If None uniform distribution is
                                     considered.
w2 : numpy array, shape (n',)
  Weight (histogram) on the samples of X2. If None uniform distribution is
                                     considered.
v1 : numpy array, shape (d,)
  Weight (histogram) on the features of X1. If None uniform distribution
                                    is considered.
v2 : numpy array, shape (d',)
  Weight (histogram) on the features of X2. If None uniform distribution
                                    is considered.
niter : integer
    Number max of iterations of the BCD for solving COOT.
algo : string
    Choice of algorithm for solving OT problems on samples each iteration.
                                     Choice ['emd','sinkhorn'].
    If 'emd' returns sparse solution
    If 'sinkhorn' returns regularized solution
algo2 : string
    Choice of algorithm for solving OT problems on features each iteration
                                    . Choice ['emd','sinkhorn'].
    If 'emd' returns sparse solution
    If 'sinkhorn' returns regularized solution
reg : float
    Regularization parameter for samples coupling matrix. Ignored if algo
                                    ='emd'
reg2 : float
```

```
    Regularization parameter for features coupling matrix. Ignored if algo
                                    ='emd'
eps : float
  Threshold for the convergence
random_init : bool
    Wether to use random initialization for the coupling matrices. If
                                    false identity couplings are
                                    considered.
log : bool, optional
   record log if True
C_lin : numpy array, shape (n, n')
    Prior on the sample correspondences. Added to the cost for the samples
                                     transport

Returns
-------
Ts : numpy array, shape (n,n')
     Optimal Transport coupling between the samples
Tv : numpy array, shape (d,d')
     Optimal Transport coupling between the features
cost : float
    Optimization value after convergence
log : dict
  convergence information and coupling marices
References
----------
.. [1] Redko Ievgen, Vayer Titouan, Flamary R{\'e}mi and Courty Nicolas
    "CO-Optimal Transport"
Example
----------
import numpy as np
from cot import cot_numpy

n_samples=300
Xs=np.random.rand(n_samples,2)
Xt=np.random.rand(n_samples,1)
cot_numpy(Xs,Xt)
"""
if v1 is None:
   v1 = np.ones(X1.shape[1]) / X1.shape[1]  # is (d,)
if v2 is None:
   v2 = np.ones(X2.shape[1]) / X2.shape[1]  # is (d',)
if w1 is None:
```

```
    w1 = np.ones(X1.shape[0]) / X1.shape[0]  # is (n',)
if w2 is None:
    w2 = np.ones(X2.shape[0]) / X2.shape[0]  # is (n,)

if not random_init:
  Ts = np.ones((X1.shape[0], X2.shape[0])) / (X1.shape[0] * X2.shape[0])
                                        # is (n,n')
  Tv = np.ones((X1.shape[1], X2.shape[1])) / (X1.shape[1] * X2.shape[1])
                                        # is (d,d')
else:
    Ts=random_gamma_init(w1,w2)
    Tv=random_gamma_init(v1,v2)


constC_s, hC1_s, hC2_s = init_matrix_np(X1, X2, v1, v2)
constC_gw, hC1_gw, hC2_gw = ot.gromov.init_matrix(C1, C2, w1, w2)
constC_v, hC1_v, hC2_v = init_matrix_np(X1.T, X2.T, w1, w2)
cost = np.inf

log_out ={}
log_out['cost'] = []

for i in range(niter):
  Tsold = Ts
  Tvold = Tv
  costold = cost

  M = constC_s - np.dot(hC1_s, Tv).dot(hC2_s.T)
  tens = ot.gromov.gwggrad(constC_gw, hC1_gw, hC2_gw, Ts)

  # print("COOT, GW SUMS, before:")
  # print(np.sum(M), np.sum(tens))

  Mmin, Mmax=np.amin(M), np.amax(M)
  tmin, tmax=np.amin(tens), np.amax(tens)
  M=(M-Mmin)/(Mmax-Mmin)
  tens=(tens-tmin)/(tmax-tmin)
  # M=M/np.sum(M)
  # tens=tens/np.sum(tens)
    # print("'
  M=((1-alpha)*M) + (alpha*tens)
```

```
    if algo == 'emd':
      Ts = ot.emd(w1, w2, M, numItermax=1e7)
    elif algo == 'sinkhorn':
      Ts = ot.sinkhorn(w1, w2, M, reg)

    M = constC_v - np.dot(hC1_v, Ts).dot(hC2_v.T)

    if C_lin is not None:
      M=M+C_lin
    if algo2 == 'emd':
      Tv = ot.emd(v1, v2, M, numItermax=1e7)
    elif algo2 == 'sinkhorn':
      Tv = ot.sinkhorn(v1,v2, M, reg2)

    delta = np.linalg.norm(Ts - Tsold) + np.linalg.norm(Tv - Tvold)
    cost = np.sum(M * Tv)

    if log:
      log_out['cost'].append(cost)

    if verbose:
      print('Delta: {0}  Loss: {1}'.format(delta, cost))

    if delta < 1e-16 or np.abs(costold - cost) < 1e-7:
      if verbose:
        print('converged at iter ', i)
      break
  if log:
    return Ts, Tv, cost, log_out
  else:
    return Ts, Tv, cost
```

7.3 Other relevant work published during Ph.D. program

- QH Tran, H Janati, N Courty, R Flamary, I Redko, **P Demetci**, R Singh. (2023). Unbalanced CO-Optimal Transport. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2023)*.

- **P Demetci**, W Cheng, G Darnell, X Zhou, S Ramachandran, L Crawford. (2021) Multi-scale genomic inference using biologically annotated neural networks. *PLOS Genetics*. 17(8): e1009754.

- R Singh, **P Demetci**, G Bonora, V Ramani, C Lee, H Fang, Z Duan, X Deng, J Shendure, C Disteche, W Stafford Noble. (2020) Unsupervised manifold alignment for single-cell multi-omics data *ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (ACM BCB)*

- B Alpay*, **P Demetci***, S Istrail, D Aguiar. (2020) Combinatorial and statistical prediction of gene expression from haplotype sequence. *Bioinformatics*. 36 (Supp-1): i194-i202. ** denotes co-first authorship.*

Bibliography

[1] D. Alvarez-Melis and T. S. Jaakkola. Gromov-wasserstein alignment of word embedding spaces. *arXiv preprint arXiv:1809.00013*, 2018.

[2] M. Amodio and S. Krishnaswamy. MAGAN: Aligning biological manifolds. 2018.

[3] C. Angermueller, S. J. Clark, H. J. Lee, I. C. Macaulay, M. J. Teng, T. X. Hu, F. Krueger, S. A. Smallwood, C. P. Ponting, T. Voet, G. Kelsey, O. Stegle, and W. Reik. Parallel single-cell sequencing links transcriptional and epigenetic heterogeneity. *Nature Methods*, 13(3):229–232, 2016. doi: 10.1038/nmeth.3728. URL `https://doi.org/10.1038/nmeth.3728`.

[4] R. Argelaguet. *Statistical methods for the integrative analysis of single-cell multi-omics data*. PhD thesis, Cambridge UK, 2020.

[5] R. Argelaguet, S. J. Clark, H. Mohammed, L. C. Stapel, C. Krueger, C.-A. Kapourani, I. Imaz-Rosshandler, T. Lohoff, Y. Xiang, C. W. Hanna, S. Smallwood, X. Ibarra-Soria, F. Buettner, G. Sanguinetti, W. Xie, F. Krueger, B. Göttgens, P. J. Rugg-Gunn, G. Kelsey, W. Dean, J. Nichols, O. Stegle, J. C. Marioni, and W. Reik. Multi-omics profiling of mouse gastrulation at single-cell resolution. *Nature*, 576(7787):487–491, 2019. doi: 10.1038/s41586-019-1825-8. URL `https://doi.org/10.1038/s41586-019-1825-8`.

[6] R. Argelaguet, A. S. E. Cuomo, O. Stegle, and J. C. Marioni. Computational

principles and challenges in single-cell data integration. *Nat. Biotechnol.*, 39 (10):1202–1215, Oct. 2021.

[7] N. Barkas, V. Petukhov, D. Nikolaeva, Y. Lozinsky, S. Demharter, K. Khodosevich, and P. V. Kharchenko. Joint analysis of heterogeneous single-cell rna-seq dataset collections. *Nature methods*, 16(8):695–698, 2019.

[8] A. Bhattacherjee, M. N. Djekidel, R. Chen, W. Chen, L. M. Tuesta, and Y. Zhang. Cell type-specific transcriptional programs in mouse prefrontal cortex during adolescence and addiction. *Nature Communications*, 10(1): 4169, Sep 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-12054-3. URL `https://doi.org/10.1038/s41467-019-12054-3`.

[9] G. Bonora, V. Ramani, R. Singh, H. Fang, D. L. Jackson, S. Srivatsan, R. Qiu, C. Lee, C. Trapnell, J. Shendure, Z. Duan, X. Deng, W. S. Noble, and C. M. Disteche. Single-cell landscape of nuclear configuration and gene expression during stem cell differentiation and x inactivation. *Genome Biology*, 22(1):279, 2021. doi: 10.1186/s13059-021-02432-w. URL `https://doi.org/10.1186/s13059-021-02432-w`.

[10] J. D. Buenrostro, B. Wu, H. Y. Chang, and W. J. Greenleaf. ATAC-seq: A method for assaying chromatin accessibility genome-wide. *Curr. Protoc. Mol. Biol.*, 109(1):21.29.1–21.29.9, Jan. 2015.

[11] J. D. Buenrostro, B. Wu, U. M. Litzenburger, D. Ruff, M. L. Gonzales, M. P. Snyder, H. Y. Chang, and W. J. Greenleaf. Single-cell chromatin accessibility reveals principles of regulatory variation. *Nature*, 523(7561):486–490, July 2015.

[12] Z. Cang and Q. Nie. Inferring spatial and signaling relationships between cells from single cell transcriptomic data. *Nature communications*, 11(1):1–13, 2020.

[13] K. Cao, X. Bai, Y. Hong, and L. Wan. Unsupervised topological alignment for single-cell multi-omics integration. *bioRxiv*, 2020.

[14] K. Cao, Y. Hong, and L. Wan. Manifold alignment for heterogeneous single-cell multi-omics data integration using Pamona. *Bioinformatics*, 08 2021. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab594. URL https://doi.org/10.1093/bioinformatics/btab594. btab594.

[15] 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, O. Fornes, T. Leung, A. Aguirre, F. Hammal, D. Schmelter, D. Baranasic, B. Ballester, A. Sandelin, B. Lenhard, K. Vandepoele, W. W. Wasserman, F. Parcy, and A. Mathelier. JASPAR 2022: the 9th release of the open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 50(D1):D165–D173, 11 2021. ISSN 0305-1048. doi: 10.1093/nar/gkab1113. URL https://doi.org/10.1093/nar/gkab1113.

[16] A. F. Chen, B. Parks, A. S. Kathiria, B. Ober-Reynolds, J. J. Goronzy, and W. J. Greenleaf. NEAT-seq: simultaneous profiling of intra-nuclear proteins, chromatin accessibility and gene expression in single cells. *Nat. Methods*, 19(5): 547–553, May 2022.

[17] K. H. Chen, A. N. Boettiger, J. R. Moffitt, S. Wang, and X. Zhuang. Spatially resolved, highly multiplexed RNA profiling in single cells. *Science*, 348(6233): aaa6090, Apr. 2015.

[18] S. Chen, B. B. Lake, and K. Zhang. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nature Biotechnology*, 37(12):1452–1457, Dec 2019. ISSN 1546-1696. doi: 10.1038/s41587-019-0290-0. URL https://doi.org/10.1038/s41587-019-0290-0.

[19] L. F. Cheow, E. T. Courtois, Y. Tan, R. Viswanathan, Q. Xing, R. Z. Tan, D. S. Q. Tan, P. Robson, L. Yuin-Han, S. R. Quake, and W. F. Burkholder. Single-cell multimodal profiling reveals cellular epigenetic heterogeneity. *Nature Methods*, 13(10):833–836, 2016.

[20] S. J. Clark, R. Argelaguet, C.-A. Kapourani, T. M. Stubbs, H. J. Lee, C. Alda-Catalinas, F. Krueger, G. Sanguinetti, G. Kelsey, J. C. Marioni, O. Stegle, and W. Reik. scNMT-seq enables joint profiling of chromatin accessibility DNA methylation and transcription in single cells. *Nat. Commun.*, 9(1):781, Feb. 2018.

[21] G. M. Cooper. *The Cell, A Molecular Approach, 2nd edition.*

[22] Z. Cui, H. Chang, S. Shan, and X. Chen. Generalized unsupervised manifold alignment. In *Advances in Neural Information Processing Systems*, pages 2429–2437, 2014.

[23] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[24] P. Datlinger, A. F. Rendeiro, C. Schmidl, T. Krausgruber, P. Traxler, J. Klughammer, L. C. Schuster, A. Kuchler, D. Alpar, and C. Bock. Pooled CRISPR screening with single-cell transcriptome readout. *Nat. Methods*, 14(3): 297–301, Mar. 2017.

[25] P. Demetci, R. Santorella, B. Sandstede, W. S. Noble, and R. Singh. Gromov-wasserstein optimal transport to align single-cell multi-omics data. *BioRxiv*, 2020.

[26] P. Demetci, R. Santorella, B. Sandstede, W. S. Noble, and R. Singh. Gromov-wasserstein optimal transport to align single-cell multi-omics data. *bioRxiv*,

2020. doi: 10.1101/2020.04.28.066787. URL https://www.biorxiv.org/content/early/2020/11/11/2020.04.28.066787.

[27] P. Demetci, R. Santorella, M. Chakravarthy, B. Sandstede, and R. Singh. SCOTv2: Single-Cell multiomic alignment with disproportionate cell-type representation. *J. Comput. Biol.*, 29(11):1213–1228, Nov. 2022.

[28] P. Demetci, R. Santorella, B. Sandstede, W. S. Noble, and R. Singh. SCOT: Single-cell multi-omics alignment with optimal transport. *J. Comput. Biol.*, 29 (1):3–18, Jan. 2022.

[29] P. Demetçi, R. Santorella, B. Sandstede, and R. Singh. Unsupervised integration of single-cell multi-omics datasets with disproportionate cell-type representation. In *Research in Computational Molecular Biology: 26th Annual International Conference, RECOMB 2022, San Diego, CA, USA, May 22–25, 2022, Proceedings*, pages 3–19. Springer, 2022.

[30] P. Demetci, Q. H. Tran, I. Redko, and R. Singh. Jointly aligning cells and genomic features of single-cell multi-omics data with co-optimal transport. *bioRxiv*, 2022. doi: 10.1101/2022.11.09.515883. URL https://www.biorxiv.org/content/early/2022/12/12/2022.11.09.515883.

[31] P. Demetci, Q. H. TRAN, I. Redko, and R. Singh. Simultaneous alignment of cells and features of unpaired single-cell multi-omics datasets with co-optimal transport. In *NeurIPS 2022 Workshop on Learning Meaningful Representations of Life*, 2022. URL https://openreview.net/forum?id=iXIyOS2dJ3.

[32] Y. Deng, M. Bartosovic, P. Kukanja, D. Zhang, Y. Liu, G. Su, A. Enninful, Z. Bai, G. Castelo-Branco, and R. Fan. Spatial-CUT&Tag: Spatially resolved chromatin modification profiling at the cellular level. *Science*, 375(6581):681–686, Feb. 2022.

[33] S. S. Dey, L. Kester, B. Spanjaard, M. Bienko, and A. van Oudenaarden. Integrated genome and transcriptome sequencing of the same cell. *Nat. Biotechnol.*, 33(3):285–289, Mar. 2015.

[34] A. Dixit, O. Parnas, B. Li, J. Chen, C. P. Fulco, L. Jerby-Arnon, N. D. Marjanovic, D. Dionne, T. Burks, R. Raychowdhury, B. Adamson, T. M. Norman, E. S. Lander, J. S. Weissman, N. Friedman, and A. Regev. Perturb-seq: Dissecting molecular circuits with scalable single-cell RNA profiling of pooled genetic screens. *Cell*, 167(7):1853–1866.e17, Dec. 2016.

[35] J. Dou, S. Liang, V. Mohanty, X. Cheng, S. Kim, J. Choi, Y. Li, K. Rezvani, R. Chen, and K. Chen. Unbiased integration of single cell multi-omics data. *bioRxiv*, 2020. doi: 10.1101/2020.12.11.422014. URL `https://www.biorxiv.org/content/early/2020/12/11/2020.12.11.422014`.

[36] M. Eisenstein. The secret life of cells. *Nature Methods*, 17(1):7–10, 2020. doi: 10.1038/s41592-019-0698-y. URL `https://doi.org/10.1038/s41592-019-0698-y`.

[37] L. Fang, Y. Li, L. Ma, Q. Xu, F. Tan, and G. Chen. GRNdb: decoding the gene regulatory networks in diverse human and mouse conditions. *Nucleic Acids Research*, 49(D1):D97–D103, 11 2020. ISSN 0305-1048. doi: 10.1093/nar/gkaa995. URL `https://doi.org/10.1093/nar/gkaa995`.

[38] R. Flamary and N. Courty. Pot python optimal transport library, 2017. URL `https://github.com/rflamary/POT`.

[39] T. Genomics. 10x genomics chromium single cell multiome atac + gene expression, 2020. URL `https://www.10xgenomics.com/products/single-cell-multiome-atac-plus-gene-expression`.

[40] T. Genomics. 10x visium spatial proteogenomics: Tissue profiling with transcriptomics and protein co-detection, 2022. URL https://www.10xgenomics.com/products/spatial-proteogenomics.

[41] F. Guo, L. Li, J. Li, X. Wu, B. Hu, P. Zhu, L. Wen, and F. Tang. Single-cell multi-omics sequencing of mouse early embryos and embryonic stem cells. *Cell Res.*, 27(8):967–988, Aug. 2017.

[42] F. Hammal, P. de Langen, A. Bergon, F. Lopez, and B. Ballester. ReMap 2022: a database of Human, Mouse, Drosophila and Arabidopsis regulatory regions from an integrative analysis of DNA-binding sequencing experiments. *Nucleic Acids Research*, 50(D1):D316–D325, 11 2021. ISSN 0305-1048. doi: 10.1093/nar/gkab996. URL https://doi.org/10.1093/nar/gkab996.

[43] T. Hashimshony, F. Wagner, N. Sher, and I. Yanai. CEL-Seq: single-cell RNA-Seq by multiplexed linear amplification. *Cell Rep.*, 2(3):666–673, Sept. 2012.

[44] Y. Hou, H. Guo, C. Cao, X. Li, B. Hu, P. Zhu, X. Wu, L. Wen, F. Tang, Y. Huang, and J. Peng. Single-cell triple omics sequencing reveals genetic, epigenetic, and transcriptomic heterogeneity in hepatocellular carcinomas. *Cell Res.*, 26(3):304–319, Mar. 2016.

[45] Y. Hu, K. Huang, Q. An, G. Du, G. Hu, J. Xue, X. Zhu, C.-Y. Wang, Z. Xue, and G. Fan. Simultaneous profiling of transcriptome and dna methylome from a single cell. *Genome Biology*, 17(1):88, May 2016. ISSN 1474-760X. doi: 10.1186/s13059-016-0950-z. URL https://doi.org/10.1186/s13059-016-0950-z.

[46] Y. Hu, Q. An, K. Sheu, B. Trejo, S. Fan, and Y. Guo. Single cell multi-omics technology: Methodology and application. *Frontiers in Cell and Developmental Biology*, 6:28, 2018. ISSN 2296-634X. doi: 10.3389/fcell.2018.00028. URL https://www.frontiersin.org/article/10.3389/fcell.2018.00028.

[47] S. Islam, U. Kjällquist, A. Moliner, P. Zajac, J.-B. Fan, P. Lönnerberg, and S. Linnarsson. Characterization of the single-cell transcriptional landscape by highly multiplex RNA-seq. *Genome Res.*, 21(7):1160–1167, July 2011.

[48] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold. Genome-wide mapping of in vivo protein-DNA interactions. *Science*, 316(5830):1497–1502, June 2007.

[49] K. Kamimoto, C. M. Hoffmann, and S. A. Morris. Celloracle: Dissecting cell identity via network inference and in silico gene perturbation. *bioRxiv*, 2020. doi: 10.1101/2020.02.17.947416. URL `https://www.biorxiv.org/content/early/2020/02/17/2020.02.17.947416`.

[50] L. V. Kantorovich. On the translocation of masses. *Dokl. Akad. Nauk. USSR (N.S.)*, 37:199–201, 1942.

[51] A. M. Klein, L. Mazutis, I. Akartuna, N. Tallapragada, A. Veres, V. Li, L. Peshkin, D. A. Weitz, and M. W. Kirschner. Droplet barcoding for single-cell transcriptomics applied to embryonic stem cells. *Cell*, 161(5):1187–1201, May 2015.

[52] D. Lähnemann, J. Köster, E. Szczurek, D. J. McCarthy, S. C. Hicks, M. D. Robinson, C. A. Vallejos, K. R. Campbell, N. Beerenwinkel, A. Mahfouz, et al. Eleven grand challenges in single-cell data science. *Genome biology*, 21(1):1–35, 2020.

[53] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

[54] G. Li, Y. Liu, Y. Zhang, N. Kubo, M. Yu, R. Fang, M. Kellis, and B. Ren. Joint profiling of dna methylation and chromatin architecture in single cells. *Nature Methods*, 16(10):991–993, Oct 2019. ISSN 1548-7105. doi: 10.1038/s41592-019-0502-z. URL `https://doi.org/10.1038/s41592-019-0502-z`.

[55] M. Liero, A. Mielke, and G. Savaré. Optimal entropy-transport problems and a new hellinger–kantorovich distance between positive measures. *Inventiones mathematicae*, 211(3):969–1117, 2018.

[56] J. Liu, Y. Huang, R. Singh, J.-P. Vert, and W. S. Noble. Jointly embedding multiple single-cell omics measurements. *BioRxiv*, page 644310, 2019.

[57] A. Ma, A. McDermaid, J. Xu, Y. Chang, and Q. Ma. Integrative methods and practical challenges for single-cell multi-omics. *Trends in Biotechnology*, 38 (9):1007–1022, 2020. ISSN 0167-7799. doi: https://doi.org/10.1016/j.tibtech.2020.02.013. URL `https://www.sciencedirect.com/science/article/pii/S0167779920300573`.

[58] I. C. Macaulay, W. Haerty, P. Kumar, Y. I. Li, T. X. Hu, M. J. Teng, M. Goolam, N. Saurat, P. Coupland, L. M. Shirley, M. Smith, N. Van der Aa, R. Banerjee, P. D. Ellis, M. A. Quail, H. P. Swerdlow, M. Zernicka-Goetz, F. J. Livesey, C. P. Ponting, and T. Voet. G&T-seq: parallel sequencing of single-cell genomes and transcriptomes. *Nat. Methods*, 12(6):519–522, June 2015.

[59] E. Z. Macosko, A. Basu, R. Satija, J. Nemesh, K. Shekhar, M. Goldman, I. Tirosh, A. R. Bialas, N. Kamitaki, E. M. Martersteck, J. J. Trombetta, D. A. Weitz, J. R. Sanes, A. K. Shalek, A. Regev, and S. A. McCarroll. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214, May 2015.

[60] A. Meissner, A. Gnirke, G. W. Bell, B. Ramsahoye, E. S. Lander, and R. Jaenisch. Reduced representation bisulfite sequencing for comparative high-resolution DNA methylation analysis. *Nucleic Acids Research*, 33(18):5868–5877, 01 2005. ISSN 0305-1048. doi: 10.1093/nar/gki901. URL `https://doi.org/10.1093/nar/gki901`.

[61] F. Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11:417–487, 2011.

[62] F. Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11(4):417–487, 2011.

[63] E. P. Mimitou, A. Cheng, A. Montalbano, S. Hao, M. Stoeckius, M. Legut, T. Roush, A. Herrera, E. Papalexi, Z. Ouyang, R. Satija, N. E. Sanjana, S. B. Koralov, and P. Smibert. Multiplexed detection of proteins, transcriptomes, clonotypes and CRISPR perturbations in single cells. *Nat. Methods*, 16(5): 409–412, May 2019.

[64] E. P. Mimitou, C. A. Lareau, K. Y. Chen, A. L. Zorzetto-Fernandes, Y. Hao, Y. Takeshima, W. Luo, T.-S. Huang, B. Z. Yeung, E. Papalexi, P. I. Thakore, T. Kibayashi, J. B. Wing, M. Hata, R. Satija, K. L. Nazor, S. Sakaguchi, L. S. Ludwig, V. G. Sankaran, A. Regev, and P. Smibert. Scalable, multimodal profiling of chromatin accessibility, gene expression and protein levels in single cells. *Nat. Biotechnol.*, 39(10):1246–1258, Oct. 2021.

[65] J. Navarro Gonzalez, A. S. Zweig, M. L. Speir, D. Schmelter, K. Rosenbloom, B. J. Raney, C. C. Powell, L. R. Nassar, N. Maulding, C. M. Lee, B. T. Lee, A. Hinrichs, A. Fyfe, J. Fernandes, M. Diekhans, H. Clawson, J. Casper, A. Benet-Pagès, G. P. Barber, D. Haussler, R. Kuhn, M. Haeussler, and W. Kent. The UCSC Genome Browser database: 2021 update. *Nucleic Acids Research*, 49(D1):D1046–D1057, 11 2020. ISSN 0305-1048. doi: 10.1093/nar/gkaa1070. URL `https://doi.org/10.1093/nar/gkaa1070`.

[66] M. Nitzan, N. Karaiskos, N. Friedman, and N. Rajewsky. Gene expression cartography. *Nature*, 576(7785):132–137, 2019.

[67] J. Packer and C. Trapnell. Single-cell multi-omics: An engine for new quantitative models of gene regulation. *Trends in Genetics*, 34(9):653–665, 2018. ISSN 0168-9525. doi: https://doi.org/10.1016/j.tig.2018.06.001. URL https://www.sciencedirect.com/science/article/pii/S0168952518301082.

[68] I. S. Peter and E. H. Davidson. *Genomic control process: development and evolution*. Academic Press, 2015.

[69] G. Peyré, M. Cuturi, and J. Solomon. Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*, pages 2664–2672, 2016.

[70] G. Peyré, M. Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[71] S. Pott. Simultaneous measurement of chromatin accessibility, dna methylation, and nucleosome phasing in single cells. *eLife*, 6:e23203, jun 2017. ISSN 2050-084X. doi: 10.7554/eLife.23203. URL https://doi.org/10.7554/eLife.23203.

[72] V. Ramani, X. Deng, R. Qiu, K. L. Gunderson, F. J. Steemers, C. M. Disteche, W. S. Noble, Z. Duan, and J. Shendure. Massively multiplex single-cell Hi-C. *Nat. Methods*, 14(3):263–266, Mar. 2017.

[73] F. J. Rang, K. L. de Luca, S. S. de Vries, C. Valdes-Quezada, E. Boele, P. D. Nguyen, I. Guerreiro, Y. Sato, H. Kimura, J. Bakkers, and J. Kind. Single-cell profiling of transcriptome and histone modifications with epidamid. *Molecular Cell*, 82(10):1956–1970.e14, 2022. ISSN 1097-2765. doi: https://doi.org/10.1016/j.molcel.2022.03.009. URL https://www.sciencedirect.com/science/article/pii/S1097276522002180.

[74] S. G. Rodriques, R. R. Stickels, A. Goeva, C. A. Martin, E. Murray, C. R. Vanderburg, J. Welch, L. M. Chen, F. Chen, and E. Z. Macosko. Slide-seq: A scalable technology for measuring genome-wide expression at high spatial resolution. *Science*, 363(6434):1463–1467, Mar. 2019.

[75] A. Rotem, O. Ram, N. Shoresh, R. A. Sperling, A. Goren, D. A. Weitz, and B. E. Bernstein. Single-cell ChIP-seq reveals cell subpopulations defined by chromatin state. *Nat. Biotechnol.*, 33(11):1165–1172, Nov. 2015.

[76] A. J. Rubin, K. R. Parker, A. T. Satpathy, Y. Qi, B. Wu, A. J. Ong, M. R. Mumbach, A. L. Ji, D. S. Kim, S. W. Cho, B. J. Zarnegar, W. J. Greenleaf, H. Y. Chang, and P. A. Khavari. Coupled single-cell CRISPR screening and epigenomic profiling reveals causal gene regulatory networks. *Cell*, 176(1-2): 361–376.e17, Jan. 2019.

[77] G. Schiebinger, J. Shu, M. Tabaka, B. Cleary, V. Subramanian, A. Solomon, J. Gould, S. Liu, S. Lin, P. Berube, et al. Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(4):928–943, 2019.

[78] A. D. Schmitt, M. Hu, and B. Ren. Genome-wide mapping and analysis of chromosome architecture. *Nat. Rev. Mol. Cell Biol.*, 17(12):743–755, Dec. 2016.

[79] R. Singh, P. Demetci, G. Bonora, V. Ramani, C. Lee, H. Fang, Z. Duan, X. Deng, J. Shendure, C. Disteche, et al. Unsupervised manifold alignment for single-cell multi-omics data. *BioRxiv*, 2020.

[80] S. A. Smallwood, H. J. Lee, C. Angermueller, F. Krueger, H. Saadeh, J. Peat, S. R. Andrews, O. Stegle, W. Reik, and G. Kelsey. Single-cell genome-wide bisulfite sequencing for assessing epigenetic heterogeneity. *Nat. Methods*, 11(8): 817–820, Aug. 2014.

[81] M. Stoeckius, C. Hafemeister, W. Stephenson, B. Houck-Loomis, P. K. Chattopadhyay, H. Swerdlow, R. Satija, and P. Smibert. Simultaneous epitope and transcriptome measurement in single cells. *Nature Methods*, 14(9):865–868, Sep 2017. ISSN 1548-7105. doi: 10.1038/nmeth.4380. URL `https://doi.org/10.1038/nmeth.4380`.

[82] T. Stuart and R. Satija. Integrative single-cell analysis. *Nat. Rev. Genet.*, 20 (5):257–272, May 2019.

[83] T. Stuart, A. Butler, P. Hoffman, C. Hafemeister, E. Papalexi, W. M. M. III, Y. Hao, M. Stoeckius, P. Smibert, and R. Satija. Comprehensive integration of single-cell data. *Cell*, 77(7):1888–1902, 2019.

[84] T. Stuart, A. Srivastava, S. Madad, C. A. Lareau, and R. Satija. Single-cell chromatin state analysis with signac. *Nature Methods*, 18(11):1333–1341, Nov 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01282-5. URL `https://doi.org/10.1038/s41592-021-01282-5`.

[85] E. Swanson, C. Lord, J. Reading, A. T. Heubeck, P. C. Genge, Z. Thomson, M. D. A. Weiss, X.-J. Li, A. K. Savage, R. R. Green, T. R. Torgerson, T. F. Bumol, L. T. Graybuck, and P. J. Skene. Simultaneous trimodal single-cell measurement of transcripts, epitopes, and chromatin accessibility using TEA-seq. *Elife*, 10, Apr. 2021.

[86] T. Séjourné, F.-X. Vialard, and G. Peyré. The unbalanced gromov wasserstein distance: Conic formulation and relaxation. *arXiv*, 2021.

[87] F. Tang, C. Barbacioru, Y. Wang, E. Nordman, C. Lee, N. Xu, X. Wang, J. Bodeau, B. B. Tuch, A. Siddiqui, K. Lao, and M. A. Surani. mrna-seq whole-transcriptome analysis of a single cell. *Nature Methods*, 6(5):377–382,

May 2009. ISSN 1548-7105. doi: 10.1038/nmeth.1315. URL https://doi.org/10.1038/nmeth.1315.

[88] V. Titouan, I. Redko, R. Flamary, and N. Courty. Co-optimal transport. *Advances in neural information processing systems*, 33:17559–17570, 2020.

[89] M. A. Tosches, T. M. Yamawaki, R. K. Naumann, A. A. Jacobi, G. Tushev, and G. Laurent. Evolution of pallium, hippocampus, and cortical cell types revealed by single-cell transcriptomics in reptiles. *Science*, 360(6391):881–888, 2018. doi: 10.1126/science.aar4237. URL https://www.science.org/doi/abs/10.1126/science.aar4237.

[90] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[91] K. Vandereyken, A. Sifrim, B. Thienpont, and T. Voet. Methods and applications for single-cell and spatial multi-omics. *Nature Reviews Genetics*, Mar 2023. ISSN 1471-0064. doi: 10.1038/s41576-023-00580-2. URL https://doi.org/10.1038/s41576-023-00580-2.

[92] T. Vayer, R. Flamary, R. Tavenard, L. Chapel, and N. Courty. Sliced gromov-wasserstein, 2022.

[93] C. Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

[94] C. Wang and S. Mahadevan. Manifold alignment without correspondence. In *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[95] Y. Wang, P. Yuan, Z. Yan, M. Yang, Y. Huo, Y. Nie, X. Zhu, J. Qiao, and L. Yan. Single-cell multiomics sequencing reveals the functional regulatory landscape of early embryos. *Nat. Commun.*, 12(1):1247, Feb. 2021.

[96] Z. Wang, M. Gerstein, and M. Snyder. RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.*, 10(1):57–63, Jan. 2009.

[97] J. D. Welch, A. J. Hartemink, and J. F. Prins. Matcher: manifold alignment reveals correspondence between single cell transcriptome and epigenome dynamics. *Genome biology*, 18(1):138, 2017.

[98] 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. *Cell*, 177(7):1873–1887, 2019.

[99] N. K. Wilson, D. G. Kent, F. Buettner, M. Shehata, I. C. Macaulay, F. J. Calero-Nieto, M. Sánchez Castillo, C. A. Oedekoven, E. Diamanti, R. Schulte, C. P. Ponting, T. Voet, C. Caldas, J. Stingl, A. R. Green, F. J. Theis, and B. Göttgens. Combined single-cell functional and gene expression analysis resolves heterogeneity within stem cell populations. *Cell Stem Cell*, 16(6):712–724, June 2015.

[100] M. Wong, C. Kosman, L. Takahashi, and N. Ramalingam. Simultaneous quantification of single-cell proteomes and transcriptomes in integrated fluidic circuits. *Methods Mol. Biol.*, 2386:219–261, 2022.

[101] X. Wu, B. Yang, I. Udo-Inyang, S. Ji, D. Ozog, L. Zhou, and Q.-S. Mi. Research techniques made simple: single-cell rna sequencing and its applications in dermatology. *Journal of Investigative Dermatology*, 138(5):1004–1009, 2018.

[102] C. Xia, J. Fan, G. Emanuel, J. Hao, and X. Zhuang. Spatial transcriptome profiling by merfish reveals subcellular rna compartmentalization and cell cycle-dependent gene expression. *Proceedings of the National Academy of Sciences*, 116(39):19490–19499, 2019. doi: 10.1073/pnas.1912459116. URL `https://www.pnas.org/doi/abs/10.1073/pnas.1912459116`.

[103] R. Yan, C. Gu, D. You, Z. Huang, J. Qian, Q. Yang, X. Cheng, L. Zhang, H. Wang, P. Wang, and F. Guo. Decoding dynamic epigenetic landscapes in human oocytes using single-cell multi-omics sequencing. *Cell Stem Cell*, 28(9): 1641–1656.e7, Sept. 2021.

[104] K. D. Yang and C. Uhler. Multi-domain translation by learning uncoupled autoencoders. *arXiv preprint arXiv:1902.03515*, 2019.

[105] K. D. Yang, K. Damodaran, S. Venkatchalapathy, A. C. Soylemezoglu, G. Shivashankar, and C. Uhler. Autoencoder and optimal transport to infer single-cell trajectories of biological processes. *bioRxiv*, page 455469, 2018.

[106] L. Zappia, B. Phipson, and A. Oshlack. Splatter: simulation of single-cell rna sequencing data. *Genome biology*, 18(1):1–15, 2017.

[107] X. Zhang, C. Xu, and N. Yosef. Simulating multiple faceted variability in single cell rna sequencing. *Nature Communications*, 10(1):2611, Jun 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-10500-w. URL `https://doi.org/10.1038/s41467-019-10500-w`.

[108] Z. Zhang, C. Yang, and X. Zhang. scDART: integrating unmatched scRNA-seq and scATAC-seq data and learning cross-modality relationship simultaneously. *Genome Biology*, 23(1):139, 2022. ISSN 1474-760X. doi: 10.1186/s13059-022-02706-x. URL `https://doi.org/10.1186/s13059-022-02706-x`.

[109] G. X. Y. Zheng, J. M. Terry, P. Belgrader, P. Ryvkin, Z. W. Bent, R. Wilson, S. B. Ziraldo, T. D. Wheeler, G. P. McDermott, J. Zhu, M. T. Gregory, J. Shuga, L. Montesclaros, J. G. Underwood, D. A. Masquelier, S. Y. Nishimura, M. Schnall-Levin, P. W. Wyatt, C. M. Hindson, R. Bharadwaj, A. Wong, K. D. Ness, L. W. Beppu, H. J. Deeg, C. McFarland, K. R. Loeb, W. J. Valente, N. G. Ericson, E. A. Stevens, J. P. Radich, T. S. Mikkelsen, B. J. Hindson, and J. H.

Bielas. Massively parallel digital transcriptional profiling of single cells. *Nat. Commun.*, 8(1):14049, Jan. 2017.

[110] C. Zhu, Y. Zhang, Y. E. Li, J. Lucero, M. M. Behrens, and B. Ren. Joint profiling of histone modifications and transcriptome in single cells from mouse brain. *Nature Methods*, 18(3):283–292, Mar 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01060-3. URL https://doi.org/10.1038/s41592-021-01060-3.