UC San Diego

UC San Diego Electronic Theses and Dissertations

Title

Interpretable Representations of Neural Circuit Dynamics Across Regions and Contexts

Permalink

https://escholarship.org/uc/item/5617r14f

ISBN

9798241645470

Author

Sristi, Ram Dyuthi

Publication Date

2026-04-09

Peer reviewed|Thesis/dissertation

eScholarship.org Powered by the California Digital Library
University of California

UNIVERSITY OF CALIFORNIA SAN DIEGO

Interpretable Representations of Neural Circuit Dynamics Across Regions and Contexts

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Electrical Engineering (Machine Learning and Data Science)

by

Ram Dyuthi Sristi

Committee in charge:

Professor Gal Mishne, Chair
Professor Mikio Aoi
Professor Vikash Gilja
Professor Takaki Komiyama
Professor Piya Pal

2026

Copyright

Ram Dyuthi Sristi, 2026

All rights reserved.

The Dissertation of Ram Dyuthi Sristi is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2026

DEDICATION

I dedicate this thesis to my wonderful family—my mother, father, and sister—whose love, encouragement, and belief in me have shaped every step of this journey.

Most of all, to my loving husband, whose patience, support, and constant faith in me carried me through the many challenges along the way.

TABLE OF CONTENTS

Dissertation Approval Page iii

Dedication iv

Table of Contents v

List of Figures viii

List of Tables xi

Acknowledgements xii

Vita xvii

Abstract of the Dissertation xviii

Chapter 1 Introduction 1
1.1 The challenge of interpreting neural population data 1
1.2 Three perspectives on neural population dynamics 3
1.3 Reorganization across conditions 5
1.4 Context dependence in neural representations 8
1.5 Distributed computation across brain regions 10
1.6 Overview of the dissertation 12

Chapter 2 DiSC: Differential Spectral Clustering 14
2.1 Motivation and background 14
2.2 Methodology for Differential Feature Grouping 17
2.2.1 Problem formulation 17
2.2.2 A graph cut perspective 18
2.2.3 Spectral approach for group differential feature extraction 20
2.2.4 Extension to multiple datasets 22
2.3 Theoretical analysis 23
2.3.1 Recovery guarantees under a stochastic block model 23
2.3.2 Proofs of recovery guarantees 26
2.3.3 Alternative justification via normalized cuts 33
2.4 Empirical evaluation and applications 35
2.4.1 Controlled toy problems 35
2.4.2 Differential functional connectivity during working memory 38
2.4.3 Differential gene structure in single-cell RNA sequencing 40
2.4.4 Differential pixel structures in handwritten digit images 41
2.4.5 Change detection in hyperspectral imagery 42
2.4.6 Hyperparameter selection 45

Chapter 3 Conditional Stochastic Gates for Contextual Feature Selection 47
3.1 Motivation for contextual feature selection 47
3.2 Problem setup and background 50
3.3 Conditional stochastic gates 51
3.4 Theoretical analysis 54
3.5 Weighted conditional stochastic gates 58
3.6 Empirical evaluation and applications 59
3.6.1 Controlled toy problems 59
3.6.2 Image classification with contextual variability 63
3.6.3 Time-dependent neuron relevance during reach outcome decoding 66
3.6.4 Flavor-dependent neuron relevance in outcome decoding 68
3.6.5 Age and gender dependent risk factors for heart disease 69
3.6.6 Location-dependent socio-economic feature relevance 70
3.6.7 Performance under controlled sparsity 71
3.6.8 Implementation details 73

Chapter 4 Coupled Transformer Autoencoder for Disentangling Multi-Region Neural Latent Dynamics 77
4.1 Motivation and related work 77
4.2 Problem formulation 80
4.3 CTAE for two-region neural recordings 81
4.3.1 Model Architecture 82
4.3.2 Training Objective 84
4.4 Extending CTAE to multi-region recordings 86
4.4.1 Problem formulation for three regions 86
4.4.2 General multi-region formulation 88
4.4.3 Multi-region training objective 89
4.5 Empirical evaluation on controlled toy dataset 91
4.6 Applications to motor cortical dynamics: M1–PMd 93
4.6.1 Dataset and behavioral task 93
4.6.2 CTAE latent dynamics in the motor circuit 95
4.6.3 Behavioral decoding from CTAE latent subspaces 97
4.6.4 Ablation study: contribution of individual loss terms 99
4.6.5 Baseline comparisons 102
4.7 Applications to multisensory circuit dynamics: SC–ALM 105
4.7.1 Dataset and multisensory task 105
4.7.2 Multi-region CTAE representative latents 107
4.7.3 Behavioral decoding from CTAE latent subspaces 108
4.7.4 Latent structure across task variables 112
4.7.5 Baseline comparisons 112
4.8 Training and Hyperparameter Details 114

Chapter 5 Complementary Cortical and Thalamic Contributions to Cell-Type-Specific Striatal Dynamics During Movement 120

5.1 Introduction 120
5.2 Experimental Design and Data Acquisition 122
5.2.1 Animal Models and Genetic Targeting 122
5.2.2 Surgical Procedures and Viral Injections 123
5.2.3 Motorized ladder task 124
5.2.4 In Vivo Calcium Imaging 125
5.2.5 Muscimol inactivation 126
5.2.6 Activity events 126
5.2.7 Histology and Anatomical Verification 126
5.3 Computational Methods 127
5.3.1 Trial Ensemble Attention Network (TEA-net) 127
5.3.2 Hierachical clustering of latent space trajectories 130
5.3.3 Forelimb and neuronal rhythmicity analysis 133
5.4 Results 135
5.4.1 Population activity patterns of dMSNs and iMSNs and their cortical and thalamic inputs 135
5.4.2 Cell-type classification based on ensembles of single-trial neuronal activity 141
5.4.3 Heterogeneous and distinctive activity patterns in DLS and input areas . 143
5.4.4 Forelimb-related rhythmicity is input- and cell-type specific 151
5.5 Discussion 155

Chapter 6 Discussion and Future Directions 158
6.1 Discussion 158
6.2 Contributions 159
6.3 Limitations 161
6.4 Future Directions 162
6.5 Conclusion 164

Bibliography 166

LIST OF FIGURES

Figure 1.1. Three complementary perspectives on neural population dynamics. 4

Figure 2.1. Differential spectral clustering overview 18

Figure 2.2. Illustration of DiSC on a controlled toy example with two datasets 36

Figure 2.3. Identifying subsets of correlated feature groups across datasets 38

Figure 2.4. Illustration of DiSC on a controlled toy example with multiple datasets . . . 39

Figure 2.5. Differential functional connectivity between 0-back and 2-back working memory tasks 40

Figure 2.6. Identifying differential gene structure in simulated scRNA-seq data 41

Figure 2.7. Hyperspectral change detection under varying acquisition conditions 44

Figure 2.8. Differential feature structures identified by DiSC in hyperspectral imagery 44

Figure 2.9. Robust hyperspectral change detection under noise 45

Figure 3.1. Contextual feature selection framework 53

Figure 3.2. Recovering context-dependent feature relevance (synthetic) 60

Figure 3.3. Learning nonlinear, weighted feature importance across contexts 63

Figure 3.4. Overcoming shrinkage in contextual feature selection 64

Figure 3.5. Context-dependent sparsity recovery in synthetic data 64

Figure 3.6. Context-dependent pixel relevance on MNIST 65

Figure 3.7. Sparsity–accuracy tradeoff under contextual feature selection 65

Figure 3.8. Sample efficiency of unified contextual models 66

Figure 3.9. Time-dependent neuron relevance during outcome decoding 67

Figure 3.10. Flavor-dependent neuron relevance during outcome decoding 68

Figure 3.11. Age and gender specific feature relevance for heart disease risk 69

Figure 3.12. Location-dependent socio-economic feature relevance in housing prices . . 71

Figure 4.1. CTAE conceptual illustration of shared and private dynamics. 81

Figure 4.2. CTAE architecture and latent disentanglement. 83

Figure 4.3. Alignment of learned and ground-truth latents on the synthetic dataset (CTAE vs DLAG). 92

Figure 4.4. Comparison of ground-truth and CTAE latent trajectories on the synthetic dataset. 92

Figure 4.5. Motor circuit dataset and task (M1–PMd). 94

Figure 4.6. Full set of CTAE latents in M1–PMd. 95

Figure 4.7. State-space views of shared and PMd-private latents (M1–PMd). 96

Figure 4.8. Decoding behavior from CTAE and DLAG latents (M1–PMd). 98

Figure 4.9. Time-resolved reach-direction decoding across subspaces (M1–PMd). 99

Figure 4.10. Alignment loss ablation: shared latent correspondence (M1–PMd). 101

Figure 4.11. Orthogonality loss ablation: shared–private disentanglement (M1–PMd). . 102

Figure 4.12. Concatenated-data baseline decoding on M1–PMd. 103

Figure 4.13. DeepCCA decoding performance on M1–PMd. 104

Figure 4.14. SC–ALM task schematic and trial timeline. 106

Figure 4.15. Representative CTAE latents in SC–ALM. 107

Figure 4.16. Decoding stimulus $\times$ target side from raw activity and CTAE subspaces. . 109

Figure 4.17. Decoding performance of mDLAG latent subspaces in the SC–ALM circuit 110

Figure 4.18. Time-resolved target-side decoding across CTAE subspaces (SC–ALM). . 111

Figure 4.19. Concatenated-data baseline decoding on SC–ALM. 113

Figure 4.20. CTAE latents averaged by target side (SC–ALM). 117

Figure 4.21. CTAE latents averaged by stimulus modality (SC–ALM). 118

Figure 5.1. Population activity patterns of dMSNs and iMSNs and their cortical and thalamic inputs 137

Figure 5.2. Example expression of mRuby-GCaMP6f and GRIN lens positioning 139

Figure 5.3. Muscimol inactivation in M1, M2, and PF abolishes movement-related striatal activity 140

Figure 5.4. TEA-net architecture and cell-type classification performance 142

Figure 5.5. Clustering reveals heterogeneous activity patterns across DLS and input regions 145

Figure 5.6. Latent representations exhibit stronger clustering tendency than raw neural activity 149

Figure 5.7. Forelimb-related rhythmic neural activity in striatal and thalamic neurons . 153

LIST OF TABLES

Table 2.1. Improved MNIST classification using differential features extracted by DiSC 42

Table 2.2. Effect of hyperparameter d on MNIST classification accuracy 46

Table 3.1. Predictive performance comparison on synthetic and image datasets 61

Table 3.2. Feature sparsity comparison on synthetic and image datasets 62

Table 3.3. Performance and feature sparsity on healthcare and socio-economic datasets 70

Table 3.4. Fixed-sparsity performance on synthetic and image datasets 72

Table 3.5. Fixed-sparsity performance on healthcare and socio-economic datasets 72

Table 4.1. Ablation of CTAE loss terms on reach-direction decoding (M1–PMd). 101

Table 4.2. DeepCCA vs. CTAE on SC–ALM pairs. 114

Table 4.3. Hyperparameter search ranges. 116

Table 4.4. Selected hyperparameters for M1–PMd, SC-ALM and Synthetic datasets. . 116

Table 5.1. Classification accuracy of classifiers for dMSN and iMSN discrimination . 143

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Gal Mishne, for her exceptional mentorship throughout my PhD. I am constantly amazed by the depth and breadth of her knowledge across so many fields and disciplines. More importantly, what has always stood out to me is how deeply she cares about her students. I have always felt that she genuinely has our best interests at heart. In situations where one path might benefit her and another might benefit her students, she consistently encourages us to choose the latter. Despite being an incredibly dedicated and hardworking researcher herself—often available late at night during the busiest quarters and deadlines—she has always reminded us to protect our well-being and maintain balance. When someone cares so deeply about your growth and well-being, striving to do your best and continually improve feels like the bare minimum. I am profoundly grateful for her guidance and support.

I am deeply grateful to my committee members, Takaki Komiyama, Vikash Gilja, Piya Pal, and Mikio Aoi, for their guidance and support throughout this journey. Working with Takaki Komiyama from the very beginning of my PhD has been a long and rewarding collaboration. During long research projects it is easy to become sidetracked by the many interesting directions that emerge along the way, but Takaki has an exceptional ability to bring us back to the bigger picture and help set the right pace. His perspective has been instrumental in shaping the direction of this work. Vikash Gilja has been a thoughtful and insightful mentor; I have always admired how calm and composed he is in every situation and how carefully he listens through an entire line of reasoning before offering clear and constructive feedback. Piya Pal had a lasting influence on how I approach research by emphasizing the importance of strong fundamentals and thoughtful reasoning, as well as the value of quality over quantity. Mikio Aoi's thoughtful questions often reveal entirely new perspectives on the same theory or framework, encouraging a broader way of thinking about research problems.

I have also been fortunate to collaborate with many wonderful researchers during my PhD. I am especially grateful to Enida Gjoni, who was a postdoctoral researcher in Takaki Komiyama's

lab and a close collaborator on the striatal calcium imaging project. I came into this work with no prior experience in calcium imaging or experimental neural data, and Enida was incredibly patient in guiding me through every aspect of the project—from the experimental setup to the details of the data—many times over, always with remarkable clarity and generosity. Our very different backgrounds—hers experimental and mine computational—turned out to be a real strength and created an opportunity for both of us to learn from each other. I am also grateful to Ariel Jaffe, my collaborator on my first computational project during my PhD, who showed immense patience while guiding an early PhD student through identifying network reorganization. I would also like to thank Ofir Lindenbaum and Hadas Benisty for their collaboration on context-specific feature selection and for their ability to identify the core concerns within complex questions with clarity and precision. My thanks also go to Simon Musall and Alice Despatin for our collaboration on multi-region neural data analysis, a project that grew out of a joint grant proposal aiming to study functional connectivity across cortex, thalamus, and striatum.

I am also grateful to the Marine Biological Laboratory for organizing the summer courses that I had the opportunity to attend. These programs introduced me to a wonderful and inspiring community of researchers, many of whom I now look forward to reconnecting with at conferences. Those interactions are always energizing and meaningful to me. I would especially like to thank Noga Mudrik, whom I met through these programs. Over the years we developed a small tradition of exchanging souvenirs whenever we meet at conferences, a simple but joyful ritual that I always look forward to. I am also thankful to my undergraduate professors—Sumohana Channapaya, Shantanu Desai, Amit Acharya, and Sri Rama Murty Kodukula—who first introduced me to research and helped shape my curiosity and academic direction. I would also like to thank my mentors—Vivek Sharma, Uma Harathi, and Raghavendra Sivapuram—who supported me at various stages of my journey. Their guidance and encouragement played an important role in shaping my academic foundation and approach to learning, and I remain deeply grateful for their influence.

I have been fortunate to work alongside wonderful labmates, each of whom brought

something unique to the lab and from whom I learned a great deal simply by observing how they approach research. Dhruv and I share a very special bond. In the early phase of my PhD, right after COVID, it was mostly just the two of us in the lab. We brainstormed ideas over lunch and often went on short walks whenever we got stuck, returning with fresh perspectives that helped move our work forward. Changhao taught me how to think clearly about writing and rebuttals, especially how to respond to feedback in a structured way. Sowmya brings incredible energy to the lab and is always looking out for others. Sophia consistently provides thoughtful and insightful feedback, Jesse inspires me with his focus and productivity, Chester impresses me with his intelligence and humility, and Victoria amazes me with how organized and systematic she is. I am also glad that our lab moved from SDSC to the new Data Science building, where sharing space with Sam, Tristan, Lucas, and Riley created a motivating environment and even encouraged me to take better care of my physical health.

Many friendships helped sustain me throughout this journey. I am grateful to my friends here at UC San Diego, Anivash, Manish, Ravi, Akhil, and Yeswanth, who helped me navigate the cultural transition of moving to a new place and made this city feel like home. I am also thankful for my roommate Shreya, with whom I shared four years of conversations, laughter, and everyday routines that made life feel grounded and comforting. I am deeply grateful to my close friends Sukrutha, Pranavi, Harshith, Sharath, and Hemanth, whose unwavering support gave me the confidence to aim high and keep pushing forward. I would also like to thank my many friends over the years—Sukrutha, Pranavi, Reshma, Keerthana, Sushma, Nirupama, Nikitha, Gopi, Jaswanthi, Akhil, Niyathi, Shashikanth, Sowjanya, Sharath, Harshith, Hemanth, Ashish, Ankur, Srikar, Nitya, Harsha, Rahul, Yeswanth, Uday, Ajay, Tejesh, Mahitha, and Harish. They have been a constant source of support, helping me maintain my energy, enthusiasm, and mental well-being through all phases of this journey. Whether it was game nights, spontaneous plans, or just being there when it mattered, they made this experience far more joyful, balanced, and meaningful.

Finally, I would like to thank my family. My mother's journey has been incredibly

inspiring to me. She completed her schooling and undergraduate education in Telugu medium, where all subjects were taught exclusively in Telugu. Later, after marriage, she began teaching English to first-grade students and continued learning until she eventually earned both her MPhil and PhD in English literature, later teaching at the engineering level—all while raising the two of us. My father has always been a constant source of support and motivation, taking on many family responsibilities so that we could focus on our education. He is deeply grounded and approaches every situation with calm and clarity, always reminding us that "everything happens for a reason"—not as a way to dismiss challenges, but to face them with resilience, learn from them, and move forward with confidence. For years my parents traveled nearly four hours every day so that my sister and I could attend a better school. My sister has been my constant support system, someone who understands me even before I say a word. Growing up she would save her pocket money throughout the year just to buy me birthday presents, and today she is a compassionate and hardworking gynecologist. Through their actions more than their words, my family taught me that neither success nor failure is permanent and that what truly matters is continuing to grow every single day. In many ways, their journeys have shaped mine, and I feel deeply grateful to now stand alongside them, each of us having earned a doctorate. I would especially like to thank my husband, who has been by my side through every stage of this journey—including all the stressful rebuttal phases. Whenever I struggled with confusing results, he would patiently sit with me, go through everything step by step, and somehow make the problems feel less intimidating and sometimes even magical. His kindness, encouragement, and belief in me have meant more than I can express. I am also deeply grateful to my mother-in-law and father-in-law, who have always treated me like their own daughter and whose warmth and generosity made me feel truly welcomed into the family.

This dissertation was supported by funding from the Simons Foundation Pilot Award (876513SPI), the Simons Foundation Pilot Extension Award (00003245), and teaching assistantships from the Halıcıoğlu Data Science Institute (HDSI) and the Department of Computer Science and Engineering at UC San Diego.

Chapter 2, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Mishne, Gal; Jaffe, Ariel (2022). "DiSC: Differential Spectral Clustering of Features." In *Advances in Neural Information Processing Systems 35 (NeurIPS)*. The dissertation author is the primary author of this paper.

Chapter 3, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Lindenbaum, Ofir; Lifshitz, Shira; Lavzin, Maria; Schiller, Jackie; Mishne, Gal; Benisty, Hadas (2024). "Contextual Feature Selection with Conditional Stochastic Gates." In *Proceedings of the 41st International Conference on Machine Learning (ICML)*. The dissertation author is the primary author of this paper.

Chapter 4, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Narasimha, Sowmya Manojna; Huang, Jingya; Despatin, Alice; Musall, Simon; Gilja, Vikash; Mishne, Gal (2026). "Coupled Transformer Autoencoder for Disentangling Multi-Region Neural Latent Dynamics." In *Proceedings of the International Conference on Learning Representations (ICLR)*. The dissertation author is the primary author of this paper.

Chapter 5, in full, is a reprint of the following material: Gjoni, Enida*; Sristi, Ram Dyuthi*; Liu, Haixin*; Dror, Shahar; Lin, Xinlei; O'Neil, Keelin; Arroyo, Oscar M.; Hong, Sun Woo; Kim, Hannah; Liu, Jeffrey; Blumenstock, Sonja; Lim, Byungkook; Mishne, Gal†; Komiyama, Takaki† (2026). "Complementary Cortical and Thalamic Contributions to Cell Type–Specific Striatal Activity Dynamics During Movement." *Science Advances*, 12(5): eaea3935. * denotes co–first authors; # denotes co–corresponding authors. The dissertation author contributed equally to this work and co-led conceptualization, methodology development, formal analysis, and writing of the manuscript. The dissertation author led the computational modeling and analysis of neural data and contributed to the interpretation of results in close collaboration with experimental co-authors.

VITA

2018	Bachelor of Technology, Electrical Engineering Indian Institute of Technology Hyderabad
2018–2019	Qualcomm India
2019–2022	Master of Science, Electrical Engineering (Machine Learning and Data Science) University of California San Diego
2022–2026	Doctor of Philosophy, Electrical Engineering (Machine Learning and Data Science) University of California San Diego

PUBLICATIONS

"Coupled Transformer Autoencoder for Disentangling Multi-Region Neural Latent Dynamics." *International Conference on Learning Representations (ICLR)*, 2026.

"Complementary cortical and thalamic contributions to cell-type-specific striatal activity dynamics during movement." *Science Advances*, 2026.

"Contextual Feature Selection with Conditional Stochastic Gates." *International Conference on Machine Learning (ICML)*, 2024.

"DiSC: Differential Spectral Clustering of Features." In *Proceedings of the 36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

ABSTRACT OF THE DISSERTATION

Interpretable Representations of Neural Circuit Dynamics Across Regions and Contexts

by

Ram Dyuthi Sristi

Doctor of Philosophy in Electrical Engineering (Machine Learning and Data Science)

University of California San Diego, 2026

Professor Gal Mishne, Chair

Understanding how neural populations give rise to behavior requires analytical frameworks that can disentangle high-dimensional, distributed, and context-dependent neural dynamics. Modern recording technologies now enable measurements across thousands of neurons, brain regions, and experimental conditions, but extracting interpretable structure from such data remains a fundamental challenge. In particular, neural computation is rarely localized to a single region or static across conditions; instead, it emerges from distributed interactions, selective engagement of neural subpopulations, and reorganization of population structure across behavioral contexts and task demands.

This dissertation develops a set of interpretable machine-learning frameworks to study

neural population dynamics from three complementary perspectives: reorganization across conditions, context-dependent feature relevance, and distributed computation across brain regions. First, we introduce Differential Spectral Clustering (DiSC), a graph-based spectral framework for identifying groups of features whose relational structure differs across experimental conditions. DiSC reveals condition-specific reorganization of neural populations beyond changes in mean activity and provides a principled approach for detecting coordinated feature groups that differentiate behavioral or cognitive states.

Second, we present conditional Stochastic Gates (c-STG), a context-aware feature selection framework that identifies task-relevant neural subpopulations in a conditional manner. By explicitly modeling context dependence, c-STG captures how distinct subsets of neurons become selectively informative for downstream behavioral decoding under different conditions, offering an interpretable view of sparsity and relevance in neural representations.

Third, we develop the Coupled Transformer Autoencoder (CTAE) framework to disentangle shared and region-specific latent dynamics from simultaneously recorded multi-region neural data. CTAE separates common latent structure from region-unique components, enabling the study of distributed neural computation while preserving interpretability across brain areas.

In addition to these methodological contributions, this dissertation includes a neuroscience case study examining complementary cortical and thalamic contributions to cell-type-specific striatal activity during movement. Using large-scale calcium imaging data recorded across multiple brain regions under non-simultaneous experimental conditions, we introduce Trial Ensemble Attention Networks (TEA-net) as a representational probe to assess ensemble-level separability of neural dynamics. This analysis reveals distinct and complementary roles of cortical and thalamic inputs in shaping striatal population activity, highlighting cell-type-specific and region-dependent structure under realistic experimental constraints.

Together, these chapters present a unified interpretability-driven framework for analyzing neural population dynamics. By integrating graph-based reorganization, context-dependent feature selection, and multi-region representation learning, while grounding these approaches in

biological case studies, this dissertation advances principled tools for understanding distributed and context-dependent neural computation and lays the foundation for scalable, interpretable models of brain function.

Chapter 1

Introduction

1.1 The challenge of interpreting neural population data

Understanding how neural activity gives rise to behavior and brain function is a central goal of neuroscience. Across experimental systems and measurement modalities, neural activity is increasingly analyzed at the level of populations rather than individual neurons. Population-level descriptions are essential because many aspects of neural computation—such as coordination, redundancy, and structured variability—are not properties of single units, but instead emerge from interactions among groups of neurons. As a result, a central challenge in modern neuroscience is to extract interpretable structure from neural population activity in a way that meaningfully connects data to scientific questions.

While advances in statistical modeling and machine learning have enabled increasingly accurate prediction and decoding from neural data, prediction alone is rarely sufficient for scientific understanding. In many settings, the primary goal is not simply to determine whether neural activity can predict a behavioral variable or experimental condition, but to understand how neural populations are organized, which aspects of activity are informative, and how these properties change across experimental manipulations. This places interpretability, rather than predictive performance alone, at the center of population-level analysis.

Neural population data present several intrinsic challenges that complicate interpretability. Neural activity is often noisy, strongly correlated across neurons, and heterogeneous in both

magnitude and temporal structure. Population responses typically reflect the combined influence of multiple factors, including experimental conditions, task structure, internal state, and intrinsic circuit dynamics. Importantly, these influences are not always expressed through simple changes in average activity. Instead, they may appear as changes in coordination among neurons, shifts in which subsets of neurons are informative, or differences in how population structure evolves over time. Analytical approaches that focus on a single summary statistic or assume a fixed notion of relevance risk obscuring meaningful structure in the data.

A further complication is that the notion of "structure" in neural population activity is inherently multi-faceted. Depending on the scientific question, structure may refer to coordinated groups of neurons that distinguish experimental conditions, to subsets of neurons whose relevance depends on contextual variables, or to patterns of activity that are shared or distinct across different recording sites. These forms of structure reflect different underlying questions about neural computation and cannot be reduced to a single analytical objective. Consequently, methods designed to answer one type of question may be ill-suited for another.

In practice, neural datasets also come with a range of experimental constraints. Recordings may be obtained from a single brain region or from multiple regions, collected simultaneously or across separate experiments, and may involve varying population sizes and behavioral paradigms. Effective analytical approaches must therefore be flexible enough to operate under diverse data regimes while remaining faithful to the scientific questions they aim to address. Treating population analysis as a one-size-fits-all problem can lead to models that are either overly restrictive or insufficiently interpretable.

Together, these considerations highlight that neural population activity cannot be fully characterized by a single summary statistic or analytical lens. Population structure reflects a combination of coordinated activity, conditional relevance, and shared or distinct patterns that emerge under different experimental circumstances. These properties are not independent; rather, they are intertwined aspects of how neural populations organize information and support behavior. A productive approach to interpretation therefore requires embracing this complexity

and developing analyses that can illuminate different facets of population structure without reducing them to a single notion of variability.

1.2 Three perspectives on neural population dynamics

The challenges outlined in the previous section arise from a common source: neural population activity is structured in ways that are richer than can be captured by any single summary or analytical lens. Population structure reflects how neurons act together, how their relevance depends on experimental or behavioral context, and how information is organized across recording sites or circuits. These properties are intertwined, yet each highlights a different aspect of how neural populations encode and transform information. Making progress, therefore, requires explicitly acknowledging these different facets and developing ways to reason about them in a principled manner.

In this dissertation, we organize the study of neural population activity around three complementary perspectives on population dynamics (Fig. 1.1). These perspectives are not intended to define separate problems, nor to partition neural data into disjoint categories. Instead, they provide different views of the same underlying object—neural population structure—each emphasizing a particular way in which meaningful variability can manifest. Together, they offer a conceptual framework for interpreting population-level activity without reducing it to a single notion of signal or relevance.

The first perspective focuses on reorganization across conditions. Neural population structure can change when experimental conditions, task demands, or behavioral states differ. For example, neural populations in patients with a neurological disorder may exhibit altered patterns of coordination compared to healthy controls, with subsets of neurons becoming more strongly or weakly correlated under the disease condition. Importantly, such changes are not always reflected in differences in average activity at the level of individual neurons. Instead, distinctions between conditions may emerge through changes in how neurons relate to one another—for

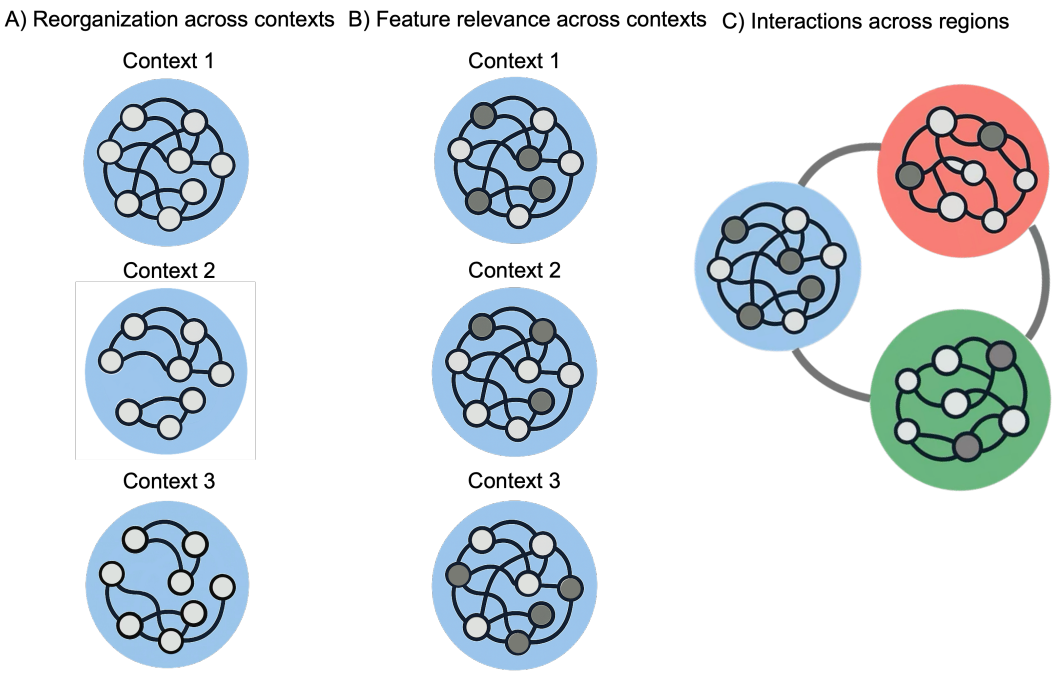


Figure 1.1. Three complementary perspectives on neural population dynamics.
(A) Reorganization across contexts: differences between experimental conditions may appear through changes in the collective organization of neural activity rather than through changes in individual neurons alone. (B) Feature relevance across contexts: the contribution of individual neurons or features to a predictive or behavioral variable can vary with context. (C) Interactions across regions: neural activity across regions may contain components that are shared across them as well as components that are region-specific.

example, through shifts in coordinated activity patterns or the formation of condition-specific groups. This perspective emphasizes population-level organization and seeks to understand how collective structure, rather than isolated features, differentiates conditions.

The second perspective addresses context dependence in neural representations. Even when the goal is to predict or decode a specific outcome, the relevance of individual neurons or features may depend on contextual variables such as task phase, stimulus configuration, or experimental parameters. For example, during a delayed reaching task, some neurons strongly encode movement direction during execution, whereas others become selectively informative during the preparatory period. In such cases, a fixed or global notion of feature importance can obscure how neural populations adapt to changing circumstances. This perspective emphasizes

conditional relevance: understanding which components of population activity matter under a given context, and how their importance varies as context changes.

The third perspective concerns distributed representations and shared structure. Neural population activity may reflect a combination of components that are common across recording sites or conditions and components that are specific to particular regions or sources. For example, during a reaching task, activity in premotor and motor cortex may share a low-dimensional component related to movement direction, while each region also exhibits region-specific dynamics reflecting its distinct role in planning and execution. Separating shared structure from source-specific activity is essential for understanding how information is coordinated and transformed, particularly when data are collected from multiple recording locations or experimental settings. This perspective highlights how population structure can be decomposed into components that support interaction and components that reflect local specialization.

Crucially, these perspectives are complementary rather than competing. Reorganization across conditions, context-dependent relevance, and shared versus specific representations describe different ways in which neural population structure and dynamics can vary. A single dataset may exhibit all three forms simultaneously, and insights gained from one perspective can inform interpretation from another. Rather than seeking a single unified description that captures all aspects of population dynamics at once, adopting multiple perspectives provides a structured way to interrogate complex neural data while preserving interpretability.

The sections that follow elaborate on each of these perspectives in turn, developing the conceptual motivation underlying each and setting the stage for the methods and analyses presented in later chapters.

1.3 Reorganization across conditions

A central question in the analysis of neural population activity is how population structure changes across experimental conditions, behavioral states, or task demands. In many settings,

differences between conditions are not fully captured by changes in the average activity of individual neurons alone. Instead, meaningful distinctions may emerge through changes in coordinated population structure, including differences in correlation patterns, functional grouping, or low-dimensional organization across subsets of neurons [Cunningham and Yu, 2014a, Vyas et al., 2020, Churchland and Shenoy, 2024]. This perspective motivates studying reorganization across conditions at the level of neural populations rather than treating neurons as independent features.

More generally, detecting features that separate between two or more conditions is a critical task across many scientific domains. Often, separation between conditions is driven not by isolated variables, but by sets of strongly dependent features that together form one or more differentiating groups. Recovering such groups can provide insight into the mechanisms underlying the data and can yield more interpretable structure than black-box discrimination alone. This idea arises in settings such as transcriptomics, where groups of genes may distinguish biological conditions or responses to treatment [Zhao et al., 2021], as well as in neuroscience and multimodal data analysis, where condition-specific structure may be expressed through coordinated relationships among features rather than marginal effects of individual variables [Xiao et al., 2018].

This viewpoint is especially relevant for neural population recordings, where correlations and shared variability are intrinsic aspects of the representation rather than nuisances to be removed [Cunningham and Yu, 2014a, Vyas et al., 2020]. From this perspective, interpretability comes not only from identifying which neurons differ across conditions, but from revealing how subsets of neurons jointly reorganize in ways that distinguish one condition from another. Such group-level structure may reflect shared inputs, circuit motifs, or coordinated population responses, and therefore offers a richer description of condition-dependent neural dynamics than feature-wise analyses alone.

An important aspect of this perspective is that reorganization across conditions need not be captured by a single differentiating group. Multiple groups of correlated features may

contribute to condition separation, potentially reflecting distinct sources of variability or different functional roles. Moreover, these groups need not be symmetric across conditions: a population pattern that is prominent in one condition may be absent, weaker, or reorganized differently in another. Capturing such asymmetric and group-specific changes is important for understanding how population structure adapts across contexts rather than reducing condition differences to a single discriminative axis.

Many commonly used analytical approaches are not ideally suited to this setting. Sparse feature selection methods often return a small subset of representative features from a larger correlated group, which can obscure the collective structure that differentiates conditions [Tibshirani, 1996b, Yamada et al., 2020]. Extensions designed to account for correlation structure, such as clustered LASSO, OWL, and elastic net, partially address this limitation but are typically tied to linear predictive formulations and do not explicitly recover multiple distinct groups of correlated features that contribute to separation [She, 2008, Bogdan et al., 2013, Figueiredo and Nowak, 2016, Zou and Hastie, 2005, Witten et al., 2014]. Likewise, discriminant subspace methods aim to identify directions that best separate classes, but do not directly target interpretable groups of dependent features [Song et al., 2010, Sharma et al., 2014, Fukunaga and Koontz, 1970, Ochilov et al., 2006, Moutafis et al., 2017]. Spectral multi-view approaches capture latent processes that appear in all modalities Lederman and Talmon [2018], Lindenbaum et al. [2020b]. In contrast, other works developed methods to recover latent processes that are unique to a single modality from the multi-modal datasets Shnitzer et al. [2019]. However, these approaches are developed for identifying shared or distinct structure between samples, not features, and they do not identify groups of samples that asymmetrically differ between condition.

These considerations motivate a complementary, data-driven view of reorganization across conditions: one that seeks to identify groups of correlated features whose relationships differ across conditions, without relying on a specific predictive model. This is the perspective taken in Chapter 2, where I develop Differential Spectral Clustering (DiSC), a spectral framework for identifying groups of features whose connectivity structure differs across conditions. In

the context of this dissertation, this perspective serves as one way of formalizing how neural population structure can reorganize across conditions, emphasizing collective structure rather than isolated feature differences.

1.4 Context dependence in neural representations

Neural population activity is often strongly shaped by contextual variables such as task condition or phase, stimulus configuration, or internal state. Rather than encoding information through a fixed set of task-relevant neurons, neural populations frequently exhibit representations in which the contribution of individual neurons depends on the behavioral or experimental context [Mante et al., 2013, Rigotti et al., 2013, Stokes et al., 2013]. Characterizing neural representations therefore requires not only identifying which signals are present in population activity, but also understanding how the relevance of different neurons varies across contexts.

From a methodological standpoint, identifying informative variables for prediction or explanation is a central objective in statistical learning and data analysis [Li et al., 2017, Kumar and Minz, 2014, Islam et al., 2022]. Most feature selection techniques, however, adopt a global perspective in which a single subset of variables is assumed to remain informative across all samples and conditions. This assumption can be restrictive in settings where the relationship between explanatory variables and outcomes is modulated by contextual factors. Context-dependent feature relevance arises in many application domains, including healthcare, recommender systems, and neuroscience, where explanatory variables may play distinct roles under different conditions [Baltrunas and Ricci, 2009, Mante et al., 2013, Rigotti et al., 2013].

Neural population recordings provide a clear illustration of this phenomenon. Activity patterns frequently exhibit mixed selectivity and high-dimensional representations, in which individual neurons respond to combinations of task variables and their functional contribution varies across behavioral contexts [Rigotti et al., 2013, Stokes et al., 2013, Fusi et al., 2016]. In such representations, task-relevant information is distributed across the population rather than

localized to a fixed subset of neurons. As a consequence, neurons that carry strong task-related signals in specific contexts may appear weak or uninformative when activity is analyzed globally across all conditions. Analytical approaches that assume a single invariant set of informative neurons can therefore obscure important aspects of population-level representation.

Existing approaches address this problem from several directions. Context-specific modeling frameworks such as Contextual Explanation Networks (CEN) [Al-Shedivat et al., 2020], Contextualized ML [Lengerich et al., 2023], and Contextual LASSO [Thompson et al., 2023] allow prediction rules to vary with contextual variables, typically by parameterizing context-dependent model coefficients. These approaches provide useful formulations of context dependence, but are often tied to linear prediction models or do not explicitly impose sparsity, limiting their ability to identify compact and interpretable sets of context-relevant features. A different line of work considers dynamic or sample-wise feature selection, in which the selected features depend on the observed sample itself [Chen et al., 2018, Yoon et al., 2018, Yang et al., 2022a], or active feature acquisition methods that iteratively choose which features to observe during inference [Shim et al., 2018, Covert et al., 2023]. While these methods permit feature relevance to vary across samples and can incorporate contextual information, they typically do not yield a direct characterization of how feature importance changes as a function of context. Recovering such structure generally requires aggregating sample-level selections across contexts, often through coarse binning or post-hoc summarization, which obscures smooth and systematic variation in relevance.

These limitations motivate a formulation in which feature relevance is modeled explicitly as a function of contextual variables. In the setting of neural population analysis, such a formulation enables determining not only which neurons are informative, but how their relevance changes across task conditions, behavioral states, or experimental settings. Chapter 3 develops this idea through Conditional Stochastic Gates (c-STG), a framework for contextual feature selection that models feature relevance as context-dependent rather than globally fixed. Within the broader scope of this dissertation, this perspective provides a way to formalize context

dependence in neural representations by focusing on conditional relevance rather than invariant feature importance.

1.5 Distributed computation across brain regions

Recent advances in large-scale neural recording technologies, including high-density electrophysiology probes and volumetric calcium imaging, have enabled simultaneous measurement of neural population activity across multiple brain regions. These recordings reveal that neural computation is rarely confined to a single brain region. Instead, behaviorally relevant signals often emerge from interactions among distributed neural populations, combining globally coordinated activity patterns with region-specific dynamics [Jun et al., 2017, Machado et al., 2022]. Understanding neural computation in such settings therefore requires analytical approaches that characterize how population activity is coordinated across interacting brain regions.

A central challenge in this setting is distinguishing activity that reflects inter-regional coordination from signals that are specific to individual populations. Neural populations in different regions can exhibit shared dynamics that mediate communication across areas while simultaneously maintaining private dynamics associated with local circuit processing. Separating these shared and region-specific components of population activity is therefore important both for understanding distributed circuit mechanisms and for interpreting perturbation experiments that manipulate specific nodes within a network.

Latent variable models provide a natural framework for studying such population-level structure. By representing high-dimensional neural activity in a lower-dimensional latent space, these models aim to capture the underlying dynamics that govern coordinated neural responses [Cunningham and Yu, 2014b]. Extending latent representations to multi-region recordings offers the possibility of jointly modeling within-region population dynamics and interactions between neural populations across different areas.

Several lines of work have developed latent-variable models for analyzing neural population activity and inter-regional interactions. Classical approaches for single-region recordings include linear dimensionality reduction methods such as PCA [Pearson, 1901] and factor analysis [Harman, 1976], as well as temporally structured extensions like Gaussian Process Factor Analysis (GPFA) [Yu et al., 2009]. More recent models introduce nonlinear dynamics through recurrent or attention-based architectures, including LFADS[Pandarinath et al., 2018] and the Neural Data Transformer [Ye and Pandarinath, 2021]. For multi-region recordings, Gaussian-process-based latent models such as DLAG [Gokcen et al., 2022] and its extensions [Gokcen et al., 2023, 2024] explicitly decompose activity into shared and region-specific components, while related approaches including multiset CCA [Hotelling, 1936], tensor decompositions [Cichocki et al., 2015], and current-based decomposition [Perich et al., 2020] have also been applied to characterize interactions across areas. In parallel, multiview representation learning methods including DCCA [Andrew et al., 2013], DCCAE [Wang et al., 2016], DMVAE [Lee and Pavlovic, 2021], MVAE [Wu and Goodman, 2018], and SPLICE [Koukuntla et al., 2024] have been proposed to disentangle shared and private latent factors across modalities or views. However, these approaches typically address only a subset of the challenges posed by large-scale multi-region neural recordings. In particular, existing methods do not simultaneously capture nonlinear and non-stationary dynamics, preserve temporal structure, and scale to recordings from more than two interacting brain regions without incurring substantial increases in model complexity.

These considerations motivate analytical approaches that explicitly model shared and region-specific representations while preserving the temporal dynamics of neural population activity. In Chapter 4, I develop the Coupled Transformer Autoencoder (CTAE), a framework for learning nonlinear latent representations that disentangle shared and private activity across multiple brain regions. Within the broader context of this dissertation, this perspective complements the earlier analyses of reorganization across conditions and context-dependent feature relevance by focusing on how neural computation emerges from interactions among distributed neural populations.

1.6 Overview of the dissertation

This dissertation develops interpretable machine learning frameworks for analyzing neural population activity from several complementary perspectives. Each chapter addresses a distinct scientific question about population-level structure—how neural activity reorganizes across conditions, how feature relevance depends on context, and how distributed brain regions coordinate activity—while emphasizing interpretable representations that remain applicable to large-scale neural datasets.

Chapter 2 addresses reorganization across experimental conditions. It introduces Differential Spectral Clustering (DiSC), a data-driven framework for identifying groups of correlated features whose relationships differ across conditions. By constructing condition-specific feature graphs and detecting changes in their connectivity structure using spectral methods, DiSC uncovers coordinated groups of differentiating features without relying on a predefined predictive model. The method is analyzed theoretically and evaluated on synthetic and real datasets—including hyperspectral imaging, single-cell RNA sequencing, and task fMRI—demonstrating its ability to recover multiple groups of condition-specific features.

Chapter 3 studies context dependence in feature relevance. It develops Conditional Stochastic Gates (c-STG), a framework for contextual feature selection in which the importance of input features varies with contextual variables that do not directly determine the prediction target. The proposed model learns context-dependent feature gates through a hypernetwork that maps contextual variables to feature-selection parameters, enabling the model to adapt its representation across contexts. Experiments on simulated and real datasets show that c-STG improves both predictive performance and interpretability in settings where feature relevance changes systematically with context.

Chapter 4 examines distributed computation across brain regions using simultaneous multi-region neural recordings. It introduces the Coupled Transformer Autoencoder (CTAE), a nonlinear latent-variable model designed to disentangle shared and region-specific neural

dynamics while capturing non-stationary temporal structure. CTAE employs Transformer-based sequence modeling to learn latent trajectories and explicitly partitions each region's latent representation into shared and private subspaces. Experiments on synthetic datasets and multi-region electrophysiology recordings demonstrate that CTAE recovers interpretable cross-region representations and improves downstream decoding of behavioral variables.

Chapter 5 presents a neuroscience study investigating how cortical and thalamic inputs shape cell-type-specific dynamics in the dorsolateral striatum during movement. Using large-scale calcium imaging data collected from identified neuronal populations under non-simultaneous experimental conditions, this chapter introduces Trial Ensemble Attention Networks (TEA-Net) as a representational probe for assessing ensemble-level separability of neural activity patterns. The resulting analyses reveal complementary contributions of cortical and thalamic pathways to striatal population dynamics, highlighting cell-type-specific and region-dependent structure in motor circuits.

Finally, Chapter 6 discusses the broader implications of these findings, highlights conceptual connections among the different methodological perspectives developed throughout the dissertation, and outlines directions for future work. Together, the chapters illustrate how interpretable machine learning frameworks can reveal structure in neural population activity across experimental conditions, behavioral contexts, and distributed brain circuits.

Chapter 2

DiSC: Differential Spectral Clustering

2.1 Motivation and background

Detecting variables or features that distinguish between two or more conditions is a fundamental task across many scientific domains. In numerous settings, differences between conditions are not driven by isolated features, but rather by groups of strongly dependent variables whose joint behavior encodes condition-specific information. Identifying such groups can provide insights into the underlying structure of the data and the mechanisms that generate it.

Strongly dependent differentiating features arise naturally in biological data. For example, in transcriptome analysis, gene expression measurements are often collected from cells belonging to different biological conditions, such as distinct cell types, disease states, responses to treatment, or different stages of development or evolution [Zhao et al., 2021]. In such settings, groups of genes—commonly referred to as pathways—may jointly distinguish between conditions, even when individual genes exhibit only weak marginal effects. Recovering these groups can substantially enhance interpretability and scientific insight.

Beyond transcriptomics, the problem of identifying groups of differentiating features appears in a wide range of applications, including neuroscience [Xiao et al., 2018], computer vision, and machine learning [Caye Daudt et al., 2018]. In all of these domains, methods that go beyond black-box prediction to reveal structured, condition-specific feature groupings are particularly valuable for interpretation and hypothesis generation.

Due to its broad relevance, feature selection has been the focus of extensive research. Many classical approaches aim to produce sparse solutions, selecting only a small subset of features from a potentially large and correlated set. One class of methods is based on subset selection, where features are iteratively added or removed according to a predefined criterion. Another widely used family of approaches incorporates regularization into predictive models, encouraging sparsity through penalty terms. For example, the LASSO replaces the non-convex ℓ_0 norm with its convex ℓ_1 surrogate [Elad, 2010, Tibshirani, 1996b], while more recent approaches approximate ℓ_0 regularization using stochastic gates [Yamada et al., 2020].

To better handle correlated features, several extensions of sparsity-inducing regularization have been proposed. Clustered LASSO [She, 2008] promotes equal coefficients within predefined groups, while Ordered Weighted ℓ_1 (OWL) regularization penalizes coefficients based on their rank by magnitude, inducing grouping among correlated variables [Bogdan et al., 2013, Figueiredo and Nowak, 2016]. Elastic Net combines ℓ_1 and ℓ_2 penalties to encourage grouped feature selection [Zou and Hastie, 2005], and Cluster Elastic Net [Witten et al., 2014] explicitly incorporates clustering penalties. Other approaches incorporate feature correlations through graph-based regularization, for example by constructing a covariance graph and using its Laplacian as a penalty term [Li et al., 2018b].

While effective in many settings, these methods typically operate within linear regression frameworks. In many applications of interest, however, the dependence on informative features is highly nonlinear, as in classification or clustering tasks. Moreover, most sparsity-based methods aim to select individual features or enforce uniform grouping, rather than explicitly separating multiple distinct groups of correlated features that contribute differently to class separation. Such separation is often crucial for understanding different sources of variability in the data.

An alternative line of work focuses on discriminant subspace analysis, including linear discriminant analysis and related methods [Song et al., 2010, Sharma et al., 2014, Fukunaga and Koontz, 1970, Ochilov et al., 2006]. These approaches aim to identify projections that maximize class separability, for example by optimizing criteria such as the Fisher ratio [Fisher, 1936].

Extensions of this idea have been explored in matrix factorization and metric learning frameworks [Gupta et al., 2014, Hess and Morik, 2017, Moutafis et al., 2017], as well as in geometry-based approaches for differential feature extraction [Cohen et al., 2022]. However, these methods generally do not explicitly recover groups of correlated features whose connectivity patterns differ across conditions.

The approach developed in this chapter is also related to work on multiview learning and data fusion. In these settings, data samples are observed through multiple modalities or sensors, and the goal is to identify shared latent structure across views. Methods such as Alternating Diffusion Maps (ADM) [Lederman and Talmon, 2018] and its extensions [Shnitzer et al., 2019, Lindenbaum et al., 2020b] recover common and view-specific structure by operating on multiple graphs. A wide range of related techniques and applications has been explored in the data fusion literature [Dong et al., 2009, Meng et al., 2020, Huang and Narayanan, 2016]. Most of these methods assume one-to-one correspondence between samples across views, for example simultaneous measurements from different sensors, and focus primarily on shared latent representations.

In contrast, the setting considered here involves correspondence between features rather than samples, with the goal of identifying feature connectivity patterns that are specific to individual conditions. Motivated by this distinction, we introduce Differential Spectral Clustering (DiSC), a data-driven method for detecting and grouping features whose connectivity structure differs across datasets. DiSC operates by constructing class-specific graphs on the feature space and applying a spectral embedding to reveal groups of correlated features that differentiate between conditions. The resulting framework is non-symmetric, enabling the identification of class-specific differential feature groups, and admits theoretical characterization under stochastic block model assumptions.

2.2 Methodology for Differential Feature Grouping

2.2.1 Problem formulation

We begin with a formal description of our problem, followed by relevant notation. We consider two datasets $X^A \in \mathbb{R}^{n_A \times p}$ and $X^B \in \mathbb{R}^{n_B \times p}$ where the rows in each matrix are high-dimensional samples and the columns are the features, see illustration in Fig. 2.1a. Both datasets have the same p features, such that the feature column in A, denoted $X^A_{\cdot i} \in \mathbb{R}^{n_A}$ corresponds to the feature column in B, denoted $X^B_{\cdot i} \in \mathbb{R}^{n_B}$. Our goal in this work is to detect one or more *subsets of features* that together differentiate between the two states A and B. These subsets can be, for example, groups of genes that form a biological pathways, or a subset of brain parcels in fMRI data with similar blood oxygen level dependent (BOLD) activity.

In our approach, we compute two graphs, denoted G_A, G_B with p nodes that correspond to the features of the given data (Fig. 2.1b). We denote by W_A, W_B the weight matrices of the two graphs, whose elements are functions of the similarity between features, as computed by two kernel functions,

$$K_A(X^A_{\cdot i}, X^A_{\cdot j}) : \mathbb{R}^{n_A} \times \mathbb{R}^{n_A} \to \mathbb{R} \qquad \text{and} \qquad K_B(X^B_{\cdot i}, X^B_{\cdot j}) : \mathbb{R}^{n_B} \times \mathbb{R}^{n_B} \to \mathbb{R}. \tag{2.1}$$

These for example can be the RBF kernel $K(x,x') = \exp\{-\|x-x'\|^2/\varepsilon^2\}$. Our underlying assumption is that differences between the two states A, B are expressed as differences in connectivity patterns between the two graphs G_A and G_B. For example, a pair of features i, j may be insignificant in state A and highly significant and dependent in B. This will imply,

$$K_A(X^A_{\cdot i}, X^A_{\cdot j}) \approx 0 \quad \text{and} \quad K_B(X^B_{\cdot i}, X^B_{\cdot j}) > 0.$$

If there is a subset of l features $i_1, \ldots, i_l$ that are strongly dependent in B but not in A, the resultant nodes in G_B will form an independent component with dense connections among the

nodes. However, this independent component will not exist in G_A, see illustration in Figure 2.1. Thus, the task of obtaining sets of significant features boils down to identifying the independent components that appear in one graph but not the other.

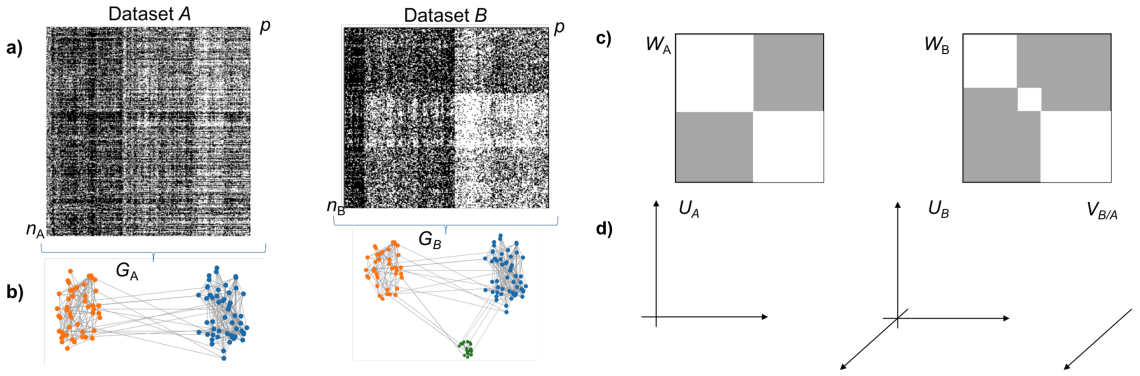


Figure 2.1. DiSC overview. a) Datasets A and B share the same p features. b) Constructing graphs G_A and G_B on the feature space for each dataset, we aim to find which nodes have different connectivity pattern between the two datasets. c) The weight matrices each follow a stochastic block model, where in B a subset of of the nodes form a separate block. d) Subspaces spanned by the eigenvectors of P_A and P_B, and the differential feature between B and A.

2.2.2 A graph cut perspective

Given a graph G with n nodes and its associated weight matrix $W \in \mathbb{R}^{n\times n}$, the minimum-cut of G is the minimum, over all possible partitions α and β, of the sum of the edge weights $\sum_{i\in\alpha, j\in\beta} W(i,j)$. This task is strongly related to the spectral clustering algorithm. For completeness, we begin with a brief description of this relation. For a more thorough review see Von Luxburg [2007].

A variation of the minimum-cut, designed to avoid highly imbalanced partitions, is the ratio-cut

$$\mathrm{Rcut}(\alpha,\beta) = \sum_{i\in\alpha, j\in\beta} W_{ij}\left(\frac{1}{|\alpha|}+\frac{1}{|\beta|}\right), \tag{2.2}$$

where $|\alpha|, |\beta|$ denote the subset size. This can be formulated as

$$\text{Rcut}(\alpha, \beta) = f^T L_u f, \qquad \text{where } f_i = \begin{cases} \sqrt{\frac{|\beta|}{|\alpha|}} & i \in \alpha \\ -\sqrt{\frac{|\alpha|}{|\beta|}} & i \in \beta, \end{cases} \tag{2.3}$$

and L_u is the unnormalized graph Laplacian matrix given by $D - W$, where D is a diagonal the degrees of each node in its diagonal. Note that f is a non-binary indicator vector where each elemnt of f, represented by f_i only takes one of the two different values as given in Eq.(2.3). If α, β are non-empty, then f is orthogonal to the constant vector with $\|f\|_2^2 = n$. Minimizing Eq. (2.3) over all partitions is a discrete optimization problem. To avoid it, one can relax the discrete requirement over the elements of f in Eq. (2.3), while still maintaining the constraints $f^T 1 = 0$ and $\|f\|_2^2 = n$. This relaxation yields a simple spectral solution, where the nodes are partitioned according to the sign of the graph Laplacian eigenvector corresponding to the second smallest eigenvalue (a.k.a Fiedler vector).

The minimum-cut criterion is designed to attain densely connected components that have low connectivity with the rest of the graph. In our work, however, densely connected components that appear in both graphs are of no interest, as they do not differentiate between the two states. Rather, we would like to detect a partition (α, β), say on G_B, that has two properties: 1) $\text{Rcut}(\alpha, \beta)$, on G_B is minimized, which indicates that α and β are independent connected components in G_B; 2) To avoid detection of components that exist both in G_A and G_B, $\text{Rcut}(\alpha, \beta)$ on G_A is lower bounded by some constant. This double objective can be formulated by

$$\min_f f^T L_B f \qquad \text{s.t.} \quad f^T L_A f \geq \gamma, \quad \|f\| = 1. \tag{2.4}$$

The constraint $\|f\| = 1$ is added for uniqueness, since L_A, L_B are rank deficient. Relaxing the discrete requirement in Eq. (2.3), the solution to Eq. (2.4) is given by the generalized eigenvalue problem $L_B f = \mu L_A f$. Since L_A is not invertible, one numerical trick is to strengthen the diagonal

of L_A to make it a full rank matrix. Relaxing the discrete requirement in (2.3), the solution is given by the eigenvalue problem,

$$(L_A + \varepsilon I)^{-1} L_B f = \mu f. \tag{2.5}$$

Motivated by the above derivation, in the next section we present our spectral approach for group feature discovery. In the appendix we present a similar derivation that is based on the normalized cut, whose relaxation relates to the eigenvectors of the random walk graph Laplacian.

2.2.3 Spectral approach for group differential feature extraction

Our proposed approach for detecting differences between graphs is motivated by the solution of the double objective criterion given in Eq. (2.5). However, due to the instability and computational complexity of the matrix inversion of the graph Laplacians L_A, L_B, we make the following two changes. First, we replace the unnormalized graph Laplacian matrices L_A and L_B with the normalized random walk Laplacians

$$P_A = D_A^{-1} W_A \qquad P_B = D_B^{-1} W_B,$$

where D_A, D_B are the degree matrices of G_A, G_B, respectively. The random-walk Laplacian is the basis of diffusion maps, which have been shown to be connected to spectral clustering in [Nadler et al., 2005, 2006]. Let $U_B \in \mathbb{R}^{p \times d_B}$ be a matrix containing the d_B leading right eigenvectors of P_B. These vectors, scaled by their corresponding eigenvalues, are the diffusion vectors of the graph G_B [Coifman and Lafon, 2006]. A vector containing the i-th element of the columns of U_A is defined as the *diffusion map representation* of the i-th node.

The second change we make is to replace the inverse operation with the following projection operator,

$$Q_A = I - U_A (U_A^T U_A)^{-1} U_A^T, \qquad Q_B = I - U_B (U_B^T U_B)^{-1} U_B^T. \tag{2.6}$$

The matrices Q_A, Q_B are projection operators to the subspace complementary to the diffusion vectors in U_A, U_B. We compute the *differential vectors* of G_A, denoted $(V_A)_i$ and their significance level $(\sigma_A)_i$ by solving the following optimization problem

$$(\sigma_A)_i = \min_{\dim E = n-i+1} \max_{v \in S(E)} ||P_A Q_B v||_2, \tag{2.7}$$

where E is a subspace in $\mathbb{R}^p$, $S(E)$ denotes unit Euclidean sphere in E, $(\sigma_A)_i$ is the optimal value of (2.7) and $(V_A)_i$ is the vector v at which optimal value of (2.7) is attained. Note that the operator $P_A Q_B$ whose singular vectors are used to attain differential features is not symmetric in A and B. The differential vectors in V_A highlight components that are significant in A but not in B. Similarly, the differential vectors V_B that are equal to the singular vectors of $P_B Q_A$ highlight components that are significant in B but not in A.

The relation between the diffusion vectors U_A and the differentiating vectors V_A is similar in nature to the relation between the outcome of the min-cut criterion in Eq. (2.3) and the double objective in Eq. (2.4). The diffusion vectors in U_B capture significant processes that underlie state B. However, we are only interested in processes that differentiate between state A and B. These are highlighted in the differential vectors V_B. In the experimental results in Sec. 2.4, we compare the diffusion maps representation to the differential vectors to illustrate this difference.

Algorithm 2.1. DiSC

Input: Datasets X^A and X^B
Two kernel functions $K_A(\cdot,\cdot)$ and $K_B(\cdot,\cdot)$
Output: Subsets of differentiating features V_A, V_B

1: Compute two graphs G_A and G_B on the columns of X^A and X^B with weights given by (2.1).
2: Compute the random walk matrix, $P_A = D_A^{-1} W_A, P_B = D_B^{-1} W_B$.
3: Calculate U_A, U_B, the leading right eigenvectors of P_A, P_B.
4: Compute the projection matrices Q_A, Q_B via Eq. (2.6).
5: Compute differential vectors V_A and V_B via Eq. (2.7).
6: Compute significance levels σ_A and σ_B via Eq. (2.7).
7: optional: Perform k-means over the rows of V_A, and V_B.

Differential vectors for downstream analysis

The differential vectors V_A, V_B and their scores σ_A, σ_B can be used in several ways. Here, and in the experimental section, we use the vectors in one of two applications: (i) Detecting of differential subsets of features. This can be done by performing k-means clustering over the rows of the matrices V_A, V_B; (ii) Computing differential meta-features via $V_A^T X_A$ and $V_B^T X_B$. In section 2.4 and supplementary material we provide several examples for using both options in downstream analysis (e.g., clustering, classification).

Computational complexity

Naively computing a graph based on a kernel function is of order $O(p^2)$. For some cases, this can be prohibitively large. However, there are simple ways to reduce the complexity to be close to linear. For example, a common approach is to compute the kernel function only for the closest k-nearest neighbors, and not for all the $O(p^2)$ pairs of features. Computing k-nearest neighbors for all features can be done effectively with structures such as KD-trees, with an average complexity (for each feature) of $k\log(p)$. Thus the total complexity of computing such a graph is $O(kplog(p))$. More efficient graph construction is possible with approximate nearest neighbors. A second advantage of this approach is that the Laplacian matrix is sparse, with only kn non zero elements. Thus, the computation of the eigenvectors can be done efficiently as well with complexity $O(kp)$.

2.2.4 Extension to multiple datasets

The extension of our approach to multiple datasets (i.e. more than two) is straightforward. We consider M datasets $X^m \in \mathbb{R}^{n_m \times p}$ for $m = 1,..,M$. We compute random-walk transition matrices on the graphs by $P_m = D_m^{-1} W_m$. Let $U_m \in \mathbb{R}^{p \times d_m}$ be a matrix containing the d_m leading right eigenvectors of P_m, and let $\hat{Q}_m$ be a projection matrix onto the the complementary subspace of $\cup_{k \neq m} U_k$, given by

$$\hat{Q}_m = I - \hat{U}_m (\hat{U}_m^T \hat{U}_m)^{-1} \hat{U}_m^T. \tag{2.8}$$

where $\hat{U}_m = [U_1, ..., U_{m-1}, U_{m+1}, ..., U_M]$. We then compute the differential vectors of G_m, denoted by $(V_m)_i$ and the corresponding significance level $(\sigma_m)_i$ using

$$(V_m)_i = \underset{\dim E = i}{\arg\max} \min_{v \in S(E)} ||P_m \hat{Q}_m v||_2. \tag{2.9}$$

$$(\sigma_m)_i = ||P_m \hat{Q}_m (V_m)_i||_2. \tag{2.10}$$

where E is a subspace in $\mathbb{R}^p$ and $S(E)$ denotes the unit Euclidean sphere in E. An algorithm implementing our approach for multiple datasets is given in Alg. 2.2.

Algorithm 2.2. DiSC - Extension to multiple datasets

Input: Datasets $X^m \in \mathbb{R}^{n_m \times p}$ for $k = 1 : M$
Kernel functions $\{K_m(\cdot,\cdot)\}_{k=1:M}$
Hyper parameters $\{d_m\}_{k=1:M}$
Output: Subsets of differentiating features $\{V_m\}_{m=1:M}$

1: Compute graphs G_m on the columns of X^m with weights given by (2.1).
2: For all the graphs, compute the random walk matrix,

$$P_m = D_m^{-1} W_m \qquad \text{for } m = 1, ..., M$$

3: Calculate $\{U_m\}_{m=1:M}$, the leading right eigenvectors of $\{P_m\}_{m=1:M}$.
4: Compute the projection matrices $\{\hat{Q}_m\}_{m=1:M}$ via Eq. (2.8).
5: Compute differential vectors $\{V_m\}_{m=1:M}$ via Eq. (2.9).
6: Compute significance levels $\{\sigma_m\}_{m=1:M}$ via Eq. (2.10).
7: optional: Perform k-means over the rows of V_A, and V_B.

2.3 Theoretical analysis

2.3.1 Recovery guarantees under a stochastic block model

The stochastic block model (SBM) has received a lot of attention due to its significant role in obtaining theoretical guarantees for community detection. In this setting, the individual community members are modeled as nodes in a random graph G. The edge weights W_{ij} are

sampled according to a Bernoulli distribution, with

$$\Pr(W_{ij}=1)=\begin{cases} p & \text{if } i,j \text{ belong to the same community} \\ q & \text{otherwise.} \end{cases} \tag{2.11}$$

Usually, one assumes $p > q$ such that the connectivity within a block is stronger than the connectivity between blocks, See [Abbe et al., 2015] for further details.

In our setting, we model the features in both states by two random graphs G_A and G_B. We consider a toy problem in which the features that differentiate two states A and B have different graph connectivity in G_A and G_B. In graph G_A the nodes are partitioned into two communities of sizes l and $l+s$, with $s < l$. In contrast, the graph G_B has three communities, denoted α, β and γ. Community α is equal to the first community in G_A, while β, γ are a partition of the second into two groups of size l and s.

Our goal is to detect the elements in γ via a spectral approach whose steps are similar to Algorithm 3.1. For simplicity of exposition, instead of using the graph Laplacian matrix as in step 2 of the algorithm, we use the symmetric weight matrices W_A, W_B. Our goal in this analysis is to provide insight into our ability to recover differentiating groups such as γ via Algorithm 3.1. Our main parameters of interest are the size of the differentiating group γ, and the ratio between the size of γ and the size of the original block. Elements that depend on other parameters of the model (e.g. p and q) are referred to as constants. Our derivation consists of three main steps. The proofs of our results are given in section 2.3.2. We note that all matrix norms (i.e. $\|X\|$) in the following section and relevant appendices are the spectral norm.

Step 1: Here we address the non-random matrix $\mathbb{E}[W_B]$ where $\mathbb{E}[\cdot]$ is the expectation operator. Let $v_\alpha, v_\beta, v_\gamma$ be the leading three eigenvectors of $\mathbb{E}[W_B]$ and by e_γ a binary indicator vector with elements $(e_\gamma)_i = 1$ if $i \in \gamma$.

Lemma 2.1. *The distance between v_γ and $\frac{1}{\sqrt{s}}e_\gamma$ is bounded by*

$$\left\| v_\gamma - \frac{1}{\sqrt{s}} e_\gamma \right\| \leq C_1(p,q)\sqrt{\frac{s}{l}}.$$

The eigenvalue corresponding to v_γ is larger than $(p-q)s$.

Lemma 2.1 shows that if the ratio s/l is small enough, one can recover the elements in γ by applying a threshold to v_γ. However, in practice we only have access to W_B and W_A. In the next step we bound the difference between v_γ and the eigenvector we compute by our spectral approach.

Step 2: Let u_α, u_β be the two leading eigenvectors of the random weight matrix W_A, and let $Q_{W_A} = I - u_\alpha u_\alpha^T - u_\beta u_\beta^T$. In this step we address the leading eigenvector of $Q_{W_A} W_B Q_{W_A}$.

Lemma 2.2. *Let $\tilde{v}_\gamma$ denote the leading eigenvector of $Q_{W_A} W_B Q_{W_A}$. Then,*

$$\|\tilde{v}_\gamma - v_\gamma\| \leq C_2(p,q)\frac{\sqrt{l}}{s} + C_3(p,q)\sqrt{\frac{s}{l}}. \qquad w.p \qquad 1-\exp(-l).$$

Step 3: Observing the two lemmas, we see that there is a tradeoff concerning the size s of the differentiating group γ. On the one hand, Lemma 2.1 shows that having a small value for s makes the element more distinguishable in v_γ and thus easier to detect. On the other hand, if $s < \sqrt{l}$ then the computed vector $\tilde{v}_\gamma$ might be too noisy to actually detect the relevant features.

Combining the two lemmas, we conclude with the following theorem.

Theorem 2.3. *We assume that s, l are large s.t. $s, l \gg \max_i(C_i)$, and $s = l^\alpha$ with $0.5 < \alpha < 1$. We apply a threshold to $\tilde{v}_\gamma$ to determine the elements of γ. The relative number of errors N_ε/s is bounded by*

$$\begin{cases} C_2(p,q) l^{1-2\alpha} & 0.5 < \alpha \leq 2/3 \\ C_3(p,q) l^{\alpha-1} & 2/3 < \alpha < 1 \end{cases}. \qquad w.p \qquad 1-\exp(-l).$$

Proof of Theorem 2.3. Let $C_4(p,q) = C_1(p,q) + C_3(p,q)$. combining Lemmas 2.1, 2.2 with the

triangle inequality, and assuming $s = l^{\alpha}$ yields the inequality

$$\left\| \tilde{v}_\gamma - \frac{1}{\sqrt{s}} e_\gamma \right\|^2 \leq C_2(p,q)^2 l^{(1-2\alpha)} + C_4(p,q)^2 l^{(\alpha-1)} + C_2(p,q) C_4(p,q) l^{-\alpha/2}.$$

For $0.5 < \alpha < 2/3$, the dominant term is $C_2(p,q)^2 l^{1-2\alpha}$. We recover the elements in γ by setting a threshold of $\frac{1}{2\sqrt{s}}$ to $\tilde{v}_\gamma$. Each misclassified element contributes at least $1/(4s)$ to the squared l_2 distance between $\frac{1}{\sqrt{s}} e_\gamma$ and $\tilde{v}_\gamma$. Thus, for $0.5 < \alpha \leq 2/3$ the relative number of errors N_ε / s is bounded with high probability by

$$\frac{N_\varepsilon}{s} \leq \left\| \tilde{v}_\gamma - \frac{1}{\sqrt{s}} e_\gamma \right\|^2 \times (4s)/s \leq 4 C_2(p,q)^2 l^{1-2\alpha}$$

A similar derivation can be done for $2/3 < \alpha < 1$. □

2.3.2 Proofs of recovery guarantees

Preliminaries

The Davis-Kahan Theorem

In our proof, we make a repeated use of the Davis-Kahan theorem. We apply both the classic theorem, and a useful variant derived in Yu et al. [2015].

Theorem 2.4 (Yu et al. [2015], Theorems 1 and 2)**.** *Let W be a symmetric matrix with eigenvectors $v_1, \ldots, v_n$ and corresponding eigenvalues $\lambda_1 \geq \lambda_2, \ldots, \lambda_n$. We denote by $\widetilde{W}$ a perturbation of W, with eigenvectors $\tilde{v}_1, \ldots, \tilde{v}_n$ and corresponding eigenvalues $\tilde{\lambda}_1 \geq \tilde{\lambda}_2, \ldots, \tilde{\lambda}_n$. Let $T_W, T_{\widetilde{W}}$ be the projection matrices onto a subspace spanned by the leading d eigenvectors,*

$$T_W = \sum_{i=1}^{d} v_i v_i^T \qquad T_{\widetilde{W}} = \sum_{i=1}^{d} \tilde{v}_i \tilde{v}_i^T .$$

In addition, we define δ via,

$$\delta = \min_{j \leq d; i > d} |\tilde{\lambda}_i - \lambda_j|.$$

Then,

$$\|T_W - T_{\widetilde{W}}\| \leq \|W - \widetilde{W}\|/\delta.$$

Alternatively,

$$\|T_W - T_{\widetilde{W}}\| \leq 2\sqrt{d}\|W - \widetilde{W}\|/(\lambda_k - \lambda_{k+1}).$$

The first inequality is the classic Davis-Kahan theorem and the second is its variant derived in Yu et al. [2015]. The importance of the variant is that it bounds the eigenvector perturbation as a function of the eigenvalues of original matrix W, with no dependency on the eigenvalues of the perturbed matrix, which are unknown in many cases. In addition, the bound on the projection matrices $T_W - T_{\widetilde{W}}$ can be replaced with a bound on the difference in norm between the subspace of eigenvectors, see for example Corollary 3 in Yu et al. [2015]. For a single vector we have

$$\|\tilde{v}_i - v_i\| \leq \frac{2^{3/2}\|W - \widetilde{W}\|}{\min(\lambda_{i-1} - \lambda_i, \lambda_i - \lambda_{i+1})}. \tag{2.12}$$

Concentration of weight matrix for stochastic block models

Another useful result is the concentration, in spectral norm, of a weight matrix generated according to the stochastic block model. This result follows directly from Bernstein's inequality for sums of independent matrices with bounded norm. This derivation is clearly presented, for example, in Vershynin [2018]

Lemma 2.5. *Let $W \in \mathbb{R}^{l\times l}$ be a matrix generated by the stochastic block model as in Eq.* (2.11)*. Then,*

$$\|W - \mathbb{E}[W]\| = C\sqrt{l} \qquad \textit{with probability} \qquad 1 - \exp(-l).$$

The rank of the expected weight matrix $\mathbb{E}[W]$ is equal to the number of communities in the model. Assume d communities and let T_W and $T_{\mathbb{E}[W]}$ denote the projection matrices onto the leading d eigenvectors of W and $\mathbb{E}[W]$ respectively. Let λ_d denote the d-th eigenvalue of $\mathbb{E}[W]$.

Combining Lemma 2.5 and Theorem 2.4 yields the following perturbation bound,

$$\|T_W - T_{\mathbb{E}[W]}\| \leq \frac{C\sqrt{dl}}{\lambda_d}. \tag{2.13}$$

where C is a constant that does not depend on the parameters of the model.

Auxiliary lemmas

For the lemmas in this subsection we have the following notation. Let $W_A, W_B \in \mathbb{R}^{(2l+s)\times(2l+s)}$ be random weight matrices obtained via the stochastic block model as described in Section 2.3.1. Let $Q_{W_A}, Q_{\mathbb{E}[W_B]}$ denote two projection matrices onto the complementary subspace of the leading eigenvectors of W_A and $\mathbb{E}[W_B]$ respectively.

Lemma 2.6. *We have the following bound on the numerator of Eq.* (2.18)*.*

$$\|Q_{W_A} W_B Q_{W_A} - Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| \leq C_1\sqrt{l} + C_2\sqrt{\frac{s}{l}}\lambda_3$$

Proof. We denote by $\mathscr{E} = Q_{W_A} - Q_{\mathbb{E}[W_B]}$. Applying the triangle inequality and the Cauchy-Schwarts inequality,

$$\begin{aligned}
&\|Q_{W_A} W_B Q_{W_A} - Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| \\
&= \|Q_{W_A} W_B (Q_{\mathbb{E}[W_B]} + \mathscr{E}) - (Q_{W_A} - \mathscr{E})\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| \\
&\leq \|Q_{W_A}(W_B - \mathbb{E}[W_B])Q_{\mathbb{E}[W_B]}\| + \|\mathscr{E}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| + \|Q_{W_A} W_B \mathscr{E}\|. \\
&\leq \|W_B - \mathbb{E}[W_B]\| + \|\mathscr{E}\|\|\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| + \|Q_{W_A} W_B\|\|\mathscr{E}\|.
\end{aligned} \tag{2.14}$$

The term $\|\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\|$ is equal by definition to the third eigenvalue of $\mathbb{E}[W_B]$. We combine lemmas 2.5, 2.7 and 2.8 to get,

$$\|Q_{W_A} W_B Q_{W_A} - Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}\| \leq C\sqrt{l} + \sqrt{\frac{s}{l}}\lambda_3 + C_2\sqrt{s}.$$

□

Lemma 2.7. *The error of* $\|\mathscr{E}\| = \|Q_{W_A} - Q_{\mathbb{E}[W_B]}\|$ *is bounded by,*

$$\|\mathscr{E}\| \leq C\sqrt{\frac{s}{l}}.$$

Proof. The triangle inequality yields,

$$\|Q_{W_A} - Q_{\mathbb{E}[W_B]}\| \leq \|Q_{W_A} - Q_{\mathbb{E}[W_A]}\| + \|Q_{\mathbb{E}[W_A]} - Q_{\mathbb{E}[W_B]}\|$$

For the first term, we use the results for the standard stochastic block model of sizes l and $l+s$. The second (and smallest non-zero) eigenvalue of $\mathbb{E}[W_A]$ is larger than $(p-q)\frac{l}{2}$

$$\|Q_{W_A} - Q_{\mathbb{E}[W_A]}\| \leq \frac{C\sqrt{l}}{(p-q)l} = \frac{C}{\sqrt{l}(p-q)}.$$

The second term bounds the difference in projection matrices of the subspace spanned by the leading two eigenvectors of $\mathbb{E}[W_A]$ and $\mathbb{E}[W_B]$. In the proof of lemma 2.1 we show that the difference between the leading eigenvectors of the two matrices is bounded by $\sqrt{\frac{s}{l}}$ and that the second eigenvector is identical. It follows that,

$$\|Q_{W_A} - Q_{\mathbb{E}[W_B]}\| \leq \frac{C}{\sqrt{l}(p-q)} + \sqrt{\frac{s}{l}}.$$

□

Lemma 2.8. *The value of* $\|Q_{W_A}W_B\|$ *is bounded by,*

$$\|Q_{W_A}W_B\| \leq C\sqrt{l}.$$

Proof. We use the triangle inequality and Cauchy Schwartz to split the term into the following,

$$\begin{aligned}\|Q_{W_A}W_B\| &\leq \|Q_{W_A}W_A\| + \|Q_{W_A}(W_A - W_B)\| \leq \|Q_{W_A}W_A\| + \|Q_{W_A}(\mathbb{E}[W_A] - \mathbb{E}[W_B])\| \\ &+ \|Q_{W_A}(W_B - \mathbb{E}[W_B])\| + \|Q_{W_A}(W_A - \mathbb{E}[W_A])\| \\ &\leq \|Q_{W_A}W_A\| + \|W_B - \mathbb{E}[W_B]\| + \|W_A - \mathbb{E}[W_A]\| + \|Q_{WA}(\mathbb{E}[W_A] - \mathbb{E}[W_B])\| \\ &\leq \|Q_{W_A}W_A\| + \|W_B - \mathbb{E}[W_B]\| + \|W_A - \mathbb{E}[W_A]\| + \|Q_{\mathbb{E}[W_A]}(\mathbb{E}[W_A] - \mathbb{E}[W_B])\| \\ &+ \|\mathbb{E}[W_A] - \mathbb{E}[W_B]\|\|Q_{\mathbb{E}[W_A]} - Q_{W_A}\|\end{aligned} \tag{2.15}$$

By lemma 2.5 (concentration of the norm of random matrix) The terms $\|W_B - \mathbb{E}[W_B]\|$ and $\|W_A - \mathbb{E}[W_A]\|$ are bounded by $C\sqrt{l}$. The term $\|Q_{W_A}W_A\|$ is equal by definition to the third eigenvalue of W_A. Recall that the third eigenvalue of $\mathbb{E}[W_A]$ is equal to zero. Thus, by weyl's inequality, the third eigenvalue of W_A is bounded by

$$\lambda_3(W_A) \leq \|W_A - \mathbb{E}[W_A]\| \leq C\sqrt{l}.$$

To bound the term $\|Q_{WA}(\mathbb{E}[W_A] - \mathbb{E}[W_B])\|$ note that the matrix $\mathbb{E}[W_A] - \mathbb{E}[W_B]$ is deterministic, and equal to

$$\mathbb{E}[W_A] - \mathbb{E}[W_B] = (p-q)e_\beta e_\gamma^T$$

with a norm of $(p-q)\sqrt{sl}$. The operator Q_{W_A} removes the average from the vector v_β over the $l+s$ last elements. Thus, the norm of $\|Q_{\mathbb{E}[W_A]}e_\beta\|$ is bounded by s/l, which implies that

$$\|Q_{\mathbb{E}[W_A]}(\mathbb{E}[W_A] - \mathbb{E}[W_B])\| \leq (p-q)s^{3/2}/l.$$

Finally, the last term is bounded by $C\frac{\sqrt{sl}}{(p-q)\sqrt{l}} = C\frac{\sqrt{s}}{p-q}$, Which is dominated by the first four terms. Summing up the different terms we have,

$$\|Q_{W_A}W_B\| \leq C\sqrt{l} + (p-q)s^{3/2}/l.$$

□

Proof of lemma 2.1

Proof. We define e_α, e_β and e_γ as the binary block indicator vectors where $(e_\alpha)_i = 1$ if $i \in \alpha$, $(e_\beta)_i = 1$ if $i \in \beta$ and $(e_\gamma)_i = 1$ if $i \in \gamma$. Let $E \in \mathbb{R}^{(2l+s)\times 3}$ be a concatenation of e_α, e_β and e_γ. We denote the pairwise block confusion matrix $\Theta \in [0,1]^{3\times 3}$ given by,

$$\Theta_{ij} = \begin{cases} p & i = j \\ q & i \neq j. \end{cases} \tag{2.16}$$

The expected weight matrix of the stochastic block model is equal to,

$$\mathbb{E}[W_B] = E\Theta E^T.$$

We denote by $\Delta \in R^{3\times 3}$ a diagonal matrix with,

$$\Delta_{11} = \sqrt{l} \quad \Delta_{22} = \sqrt{l} \quad \Delta_{33} = \sqrt{s}.$$

The expected weight matrix of the stochastic block model is equal to

$$\mathbb{E}[W_B] = E\Theta E^T = (E\Delta^{-1})(\Delta\Theta\Delta)(\Delta^{-1}E^T).$$

The matrix $E\Delta^{-1}$ is orthonormal. The eigenvalues of $\mathbb{E}[W_B]$ are thus equal to the eigenvalues of $\Delta\Theta\Delta$ and the corresponding eigenvectors are equal to $E\Delta^{-1}$ multiplied by the eigenvectors of $\Delta\Theta\Delta$. Consider the matrix Z given by,

$$Z_{ij} = \begin{cases} (\Delta\Theta\Delta)_{ij} & i,j < 3 \text{ or } i = j = 3 \\ 0 & o.w. \end{cases}$$

The eigenvectors of Z are equal to $u_1 = [1,1,0]$ and $u_2 = [1,-1,0]$ with corresponding eigenvalues $(p+q)l$ and $(p-q)l$ respectively. We denote by $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$ the eigenvectors of $\Delta\Theta\Delta$. A direct computation shows that $\tilde{u}_2 = u_2$ with the same eigenvalue. Applying Theorem 2.4 yields,

$$\|u_1 - \tilde{u}_1\| \leq \frac{2\sqrt{2}\|Z - \Delta\Theta\Delta\|}{2ql} = \frac{4\sqrt{2}q\sqrt{ls}}{2ql} = \sqrt{\frac{8s}{l}}. \tag{2.17}$$

Both u_1, u_3 and $\tilde{u}_1$, $\tilde{u}_3$ are orthogonal to $u_2 = \tilde{u}_2$ and thus span the same 2D subspace. This implies

$$\|\tilde{u}_3 - u_3\| = \|u_1 - \tilde{u}_1\| \leq \sqrt{\frac{8s}{l}}.$$

It follows directly that v_γ, the the third eigenvector of $\mathbb{E}[W_B]$ satisfies

$$\left\| v_\gamma - \frac{1}{\sqrt{s}} e_\gamma \right\| \leq \sqrt{\frac{8s}{l}}.$$

Finally, we derive a lower bound on the third eigenvalue of $\mathbb{E}[W_B]$. Recall that the eigenvalues of $\mathbb{E}[W_B]$ are equal to those of $\Delta\Theta\Delta$. For the latter we apply the inequality,

$$\lambda_{\min}(\Delta\Theta\Delta) \geq \lambda_{\min}(\Delta)\lambda_{\min}(\Theta)\lambda_{\min}(\Delta),$$

where $\lambda_{\min}$ denotes the smallest eigenvalue. The matrix Δ is diagonal with values $\sqrt{l}, \sqrt{l}, \sqrt{s}$ and hence $\lambda_{\min}(\Delta) = \sqrt{s}$. By direct computation $\lambda_{\min}(\Theta) \geq (p-q)$. Hence,

$$\lambda_3(\mathbb{E}[W_B]) = \lambda_{\min}(\Delta\Theta\Delta) \geq s(p-q).$$

□

Proof of Lemma 2.2

In this lemma we bound the difference between the vector $\tilde{v}_\gamma$ and the corresponding eigenvector of $\mathbb{E}[W_B]$, denoted by v_γ. Recall the definition of $Q_{\mathbb{E}[W_B]}$ as the projection matrix onto

the complementary subspace of the two leading eigenvectors of $\mathbb{E}[W_B]$. By definition, v_γ is the leading eigenvector of $Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]}$. Thus, our goal is to bound the leading eigenvector of two matrices:

$$Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]} \quad \text{and} \quad Q_{\mathbb{E}[W_A]}W_B Q_{\mathbb{E}[W_A]}.$$

To that end, we apply the Davis-Kahan theorem. Let λ_γ be the third eigenvector of $\mathbb{E}[W_B]$. The theorem gives the following bound.

$$\|v_\gamma - \tilde{v}_\gamma\| \leq 2^{3/2}\frac{\|Q_{\mathbb{E}[W_B]}\mathbb{E}[W_B]Q_{\mathbb{E}[W_B]} - Q_{\mathbb{E}[W_A]}W_B Q_{\mathbb{E}[W_A]}\|}{\lambda_3}. \tag{2.18}$$

Applying lemma 2.6 we get:

$$\|v_\gamma - \tilde{v}_\gamma\| \leq C\frac{\sqrt{l}}{\lambda_3} + C_2\sqrt{\frac{s}{l}}$$

For the first term, we apply the lower bound $\lambda_3 \geq (p-q)s$. This yields,

$$\|v - \tilde{v}\| \leq C_1\frac{\sqrt{l}}{s} + C_2\sqrt{\frac{s}{l}}$$

2.3.3 Alternative justification via normalized cuts

A different variation of the graph-cut criterion in (2.2) is the Normalized cut. Let d_i denote the degree of node i, and let $\text{vol}(\alpha) = \sum_i d_i$ be the sum of degrees of a subset of nodes α. The normalized cut is equal to,

$$\text{Ncut}(\alpha,\beta) = \sum_{i\in\alpha, j\in\beta} W(i,j)\Big(\frac{1}{\text{vol}(\alpha)} + \frac{1}{\text{vol}(\beta)}\Big).$$

Similarly to (2.3), we can define an indicator vector by,

$$f_i = \begin{cases} \sqrt{\frac{\text{vol}(\beta)}{\text{vol}\alpha}} & i \in \alpha \\ \sqrt{\frac{\text{vol}(\alpha)}{\text{vol}\beta}} & i \in \beta. \end{cases} \tag{2.19}$$

The normalized cut can be expressed as,

$$\text{Ncut}(\alpha, \beta) = f^T L f.$$

Here, the indicator vector is orthogonal to $D\mathbf{1}$, where D is a diagonal matrix containing the degrees $\{d_i\}$ in its diagonal and $\mathbf{1}$ is the all ones vector. In addition, though the norm of f depends on α, the term $f^T D f$ is a constant. Relaxing the requirement in Eq. (2.19), yields the following optimization problem for the normalized cut,

$$\min_f f^T L f \qquad \text{subject to} \qquad f^T D f = 1.$$

In our work, we consider two graph Laplacians L_A, L_B. For simplicity we assume that the degree matrix of both graphs is identical. Thus, the indicator vector in Eq. (2.19) is the same for both graphs and for any partition. The second graph L_B yields a second constraint on the optimization problem.

$$\min f^T L_A f \quad \text{subject to} \quad f^T D f = 1, \quad f^T L_B f = \gamma. \tag{2.20}$$

The solution to Eq. (2.20) satisfies the generalized eigenvector problem

$$L_A f = \lambda_1 (D + \lambda_2 L_B) f.$$

Multiplying both sides by D^{-1} yields,

$$(I + \lambda_2 P_b)^{-1} P_A f = \lambda_1 f, \tag{2.21}$$

where P_A, P_B are the random walk Laplacian matrices. Note that the term $(I + \lambda_2 D^{-1} L_b)^{-1}$ is a *regularized inverse* matrix of P_B, often used to ovoid an arbitrary increase of noise when computing the inverse of ill condition matrices. Thus, the expression in (2.21) is very similar in nature to the expression in Eq. 2.5.

2.4 Empirical evaluation and applications

2.4.1 Controlled toy problems

We demonstrate the behavior of DiSC using three toy examples that capture different patterns of feature correlation across datasets: (1) a subset of features that is uncorrelated in one dataset becomes correlated in another; (2) a correlated feature group in one dataset splits into multiple correlated subgroups in another; and (3) an extension to more than two datasets, where distinct feature connections are specific to individual datasets. Unless otherwise noted, all experiments use RBF kernels with adaptive bandwidth, and the dimensionalities d_A, d_B (and d_C where applicable) are chosen consistently across datasets. Details of hyperparameter selection, including kernel bandwidth and eigenspace dimensionality, are provided in Section 2.4.6.

Identifying newly connected features

We generated two datasets X^A and X^B with $p = 250$ features and $n_A = n_B = 10,000$ samples from a Gaussian mixture model (Fig. 2.2(a)), whose feature correlation is shown in Fig. 2.2(b). In the two datasets, the first 100 features are i.i.d samples from a normal distribution and the next 50 features are sampled from two Gaussian distributions with low rank covariance matrices. Features 151-200 in X^A are sampled from two other Gaussian distributions with low rank covariance matrices, whereas in X^B, they are independent noise. Features 201-250 in X^A are independent noise, whereas in X^B, they are sampled from two other Gaussian distributions with low rank covariance matrices.

Our goal is to identify the features 151-200 as the features that distinguish X^A from X^B, and vice versa for features 201-250. We apply DiSC with $d_A = d_B = 20$. The first two

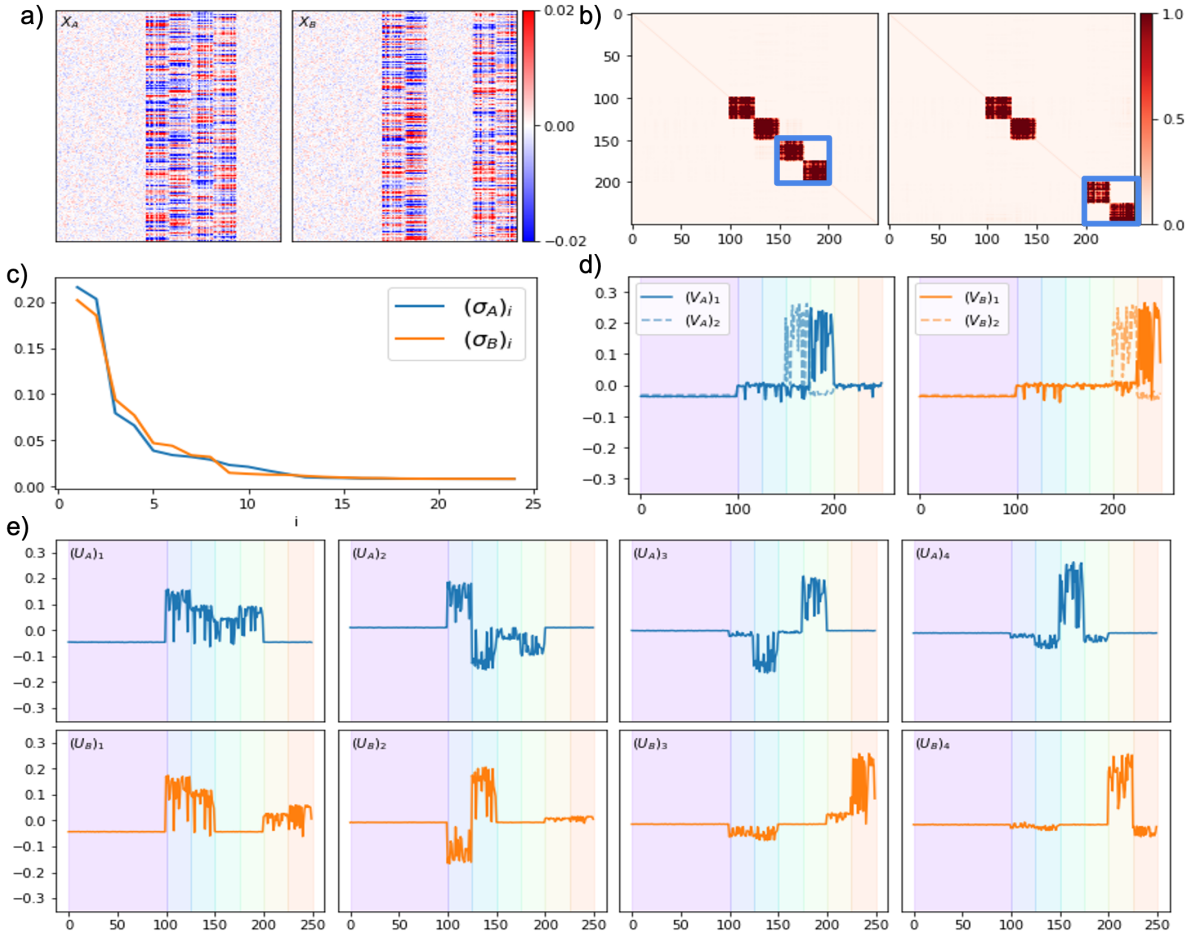


Figure 2.2. a) Random samples from X^A and X^B. b) Feature correlation matrices of X^A (left) and X^B (right). c) Significance of the differential vectors of X^A (blue) and X^B (orange). d) Top two differential vectors. e) Top four diffusion maps of the features of X^A (blue) and X^B (orange).

significance values of the differential features are high and then significance drop drastically (Fig. 2.2(c)), indicating that the first two differential vectors are important in each dataset. These top two differential features for X^A and X^B are shown in Figure 2.2(d). Clearly, $(V_A)_1$ and $(V_A)_2$ highlight the features between 151-200 and more precisely, the two Gaussian mixtures are separately represented in each of these vectors. Similarly, $(V_B)_1$ and $(V_B)_2$ highlight the features between 201-250 and these represent the other two Gaussian mixtures. On the other hand, the diffusion maps eigenvectors, U_A and U_B, captures all the connected components and not just the differential features as unique groups, as shown in Fig. 2.2(e).

Identifying subsets of connected features

We generated two datasets X^A and X^B with $p = 200$ features and $n_A = n_B = 10,000$ with correlation between the features as shown in Fig. 2.3(a). In both the datasets, the first 100 features are correlated. In X^A, all the remaining features form a second correlated group, yielding two connected components in the feature space. In X^B the remaining features are composed of two groups of correlated features, namely, feature indices 101-125 and 126-200.

Here the goal is to identify these smaller subsets of correlated features in X^B, as this signifies an increase in the dimensionality of correlation structure compared to X^A. This also means that X^B has richer feature information than X^A, therefore we have to identify that there are no differential features in X^A.

We use the significance level associated with the differential vectors to determine if the differential vectors are meaningful. We apply DiSC with $d_A = d_B = 20$. Fig. 2.3(c) shows the differential vectors of X^A and X^B and the corresponding significance levels are shown in Fig. 2.3(b). The significance level associated with the differential vectors of X^A is negligible compared to that of X^B, indicating that feature information in X^A is contained in X^B. In addition, the difference between the significance of the first two differential vectors of X^B is large. Therefore, the first differential vector of X^B captures the major difference, and it groups feature indices 101-125 and 126-200 with positive and negative values respectively (Fig. 2.3(c)). In contrast, diffusion maps captures the two connected components in X^A and three connected components in X^B, and not just the differential components (Fig. 2.3(d)).

Extending DiSC to multiple datasets

To illustrate the extension of DiSC beyond pairwise comparisons, we consider three datasets X_A, X_B, and X_C, each consisting of $p = 400$ features and $n_A = n_B = n_C = 10{,}000$ samples. The feature correlation structure in each dataset is shown in Fig. 2.4(a). Some feature connections are shared across datasets, while others are specific to individual datasets and are highlighted for clarity.

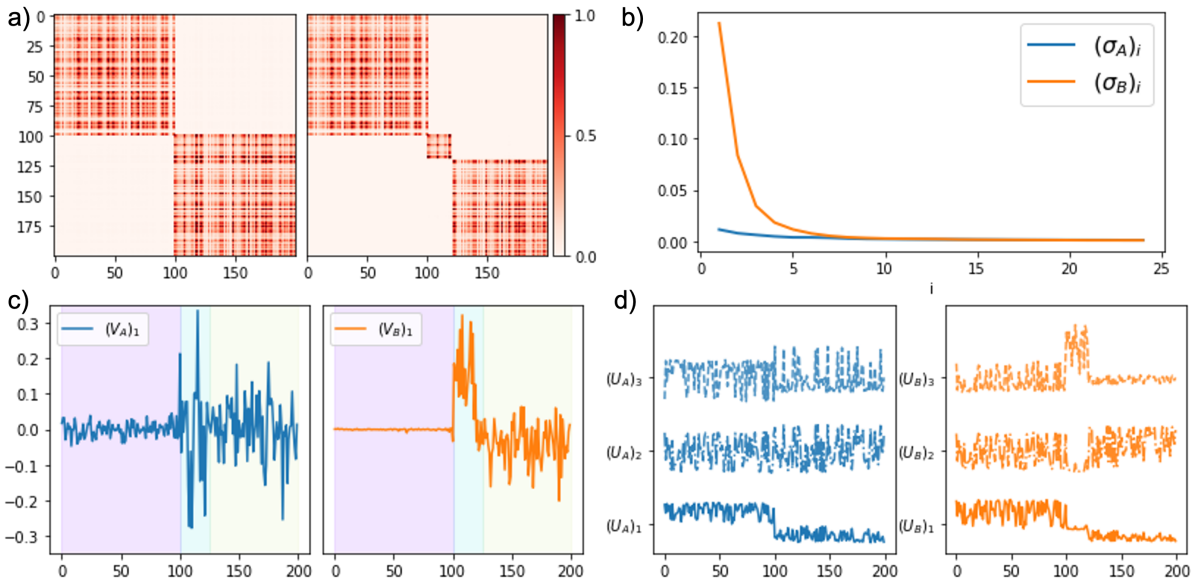


Figure 2.3. a) Correlation matrices. b) Significance levels. c) Differential features. d) Diffusion maps.

In this example, feature connections between indices 301–350 are specific to X_A, connections between 201–250 are specific to X_B, and connections between 351–400 are specific to X_C. Each dataset therefore contains two groups of correlated features that are dataset-specific, and the goal is to identify these groups individually. We apply DiSC with $d_A = d_B = d_C = 20$.

The significance levels of the resulting differential vectors are shown in Fig. 2.4(b). The first two differential vectors exhibit comparable significance across all three datasets, followed by a sharp drop, indicating that these vectors capture the dominant dataset-specific differences. As shown in Fig. 2.4(c), each differential vector isolates a distinct group of dataset-specific feature connections. In contrast, diffusion maps (Fig. 2.4(d)) capture global connectivity structure but do not separate dataset-specific feature groups as clearly.

2.4.2 Differential functional connectivity during working memory

We assess the performance of DiSC in identifying groups of brain parcels with correlated BOLD activity in a working memory task from the Human Connectome Project Van Essen et al. [2013], where subjects executed interleaved blocks of 0-back and 2-back working memory

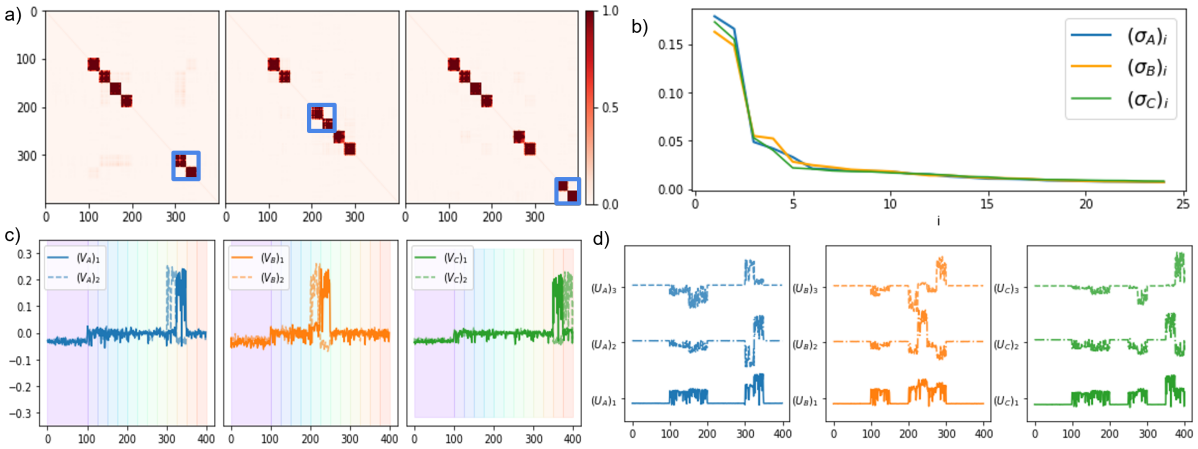


Figure 2.4. Multiple datasets. a) Correlation matrices. b) Significance levels. c) Differential features. d) Diffusion maps.

tasks. In these tasks, subjects are instructed to monitor a sequence of visual items and to respond whenever a presented item is the same as the one previously presented 2 items ago (2-back) or a predetermined item (0-back). This fMRI dataset consists of 515 subjects (subjects with high motion or incomplete data were removed), and a whole-brain, functional atlas Shen et al. [2013] was used to extract time-courses from $p = 268$ brain parcels. Dataset X^A is composed of all blocks from the 0-back task and X^B is composed of all blocks from the 2-back task.

For each subject we calculated the top 2 diffusion maps eigenvectors as well as the top 2 differential features for each dataset, and average these across all subjects. Fig. 2.5 displays the correlation of each of the averaged vectors with indicator vectors for 10 canonical brain networks Finn et al. [2015]. Results show that diffusion maps is mainly correlated with visual networks, while the differential vector for the 0-back task being most correlated with the visual II network. On the other hand, the differential vector for the 2-back task is most correlated with the frontal-parietal network, which has been shown to be predictive of working memory performance Avery et al. [2020]. Thus, as opposed to diffusion maps, DiSC reveals that the 2-back task incorporates more high-level cognitive regions (e.g., prefrontal) compared to the 0-back task which has lower cognitive load Gao et al. [2019, 2021].

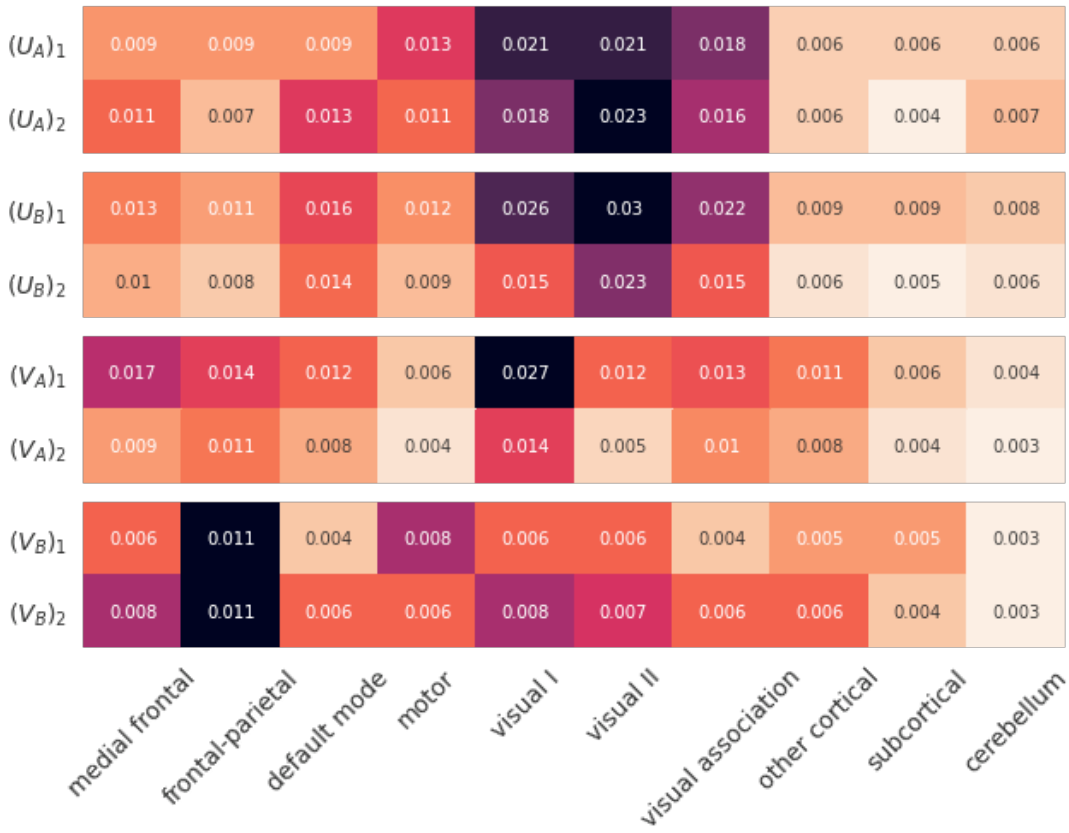


Figure 2.5. Canonical correlations of subject-averaged diffusion maps and differential features with 10 canonical brain networks for fMRI data of a 0-back (task A) and 2-back (task B) working memory tasks.

2.4.3 Differential gene structure in single-cell RNA sequencing

Using the *splatter* simulator [Zappia et al., 2017] we generated a simulated dataset containing the RNA expression level of $p = 500$ genes, as measured in 1000 cells, which belong to two different cell types. Out of the 500 genes, only approximately 100 genes are *differential*, and thus have a different expression level for the two cell types.

We visualize the cells in 2D using t-SNE [Van der Maaten and Hinton, 2008]. Figure 2.6 colors the cells by *meta features*, that are equal to weighted sum over the gene expression profile of each cell. In the upper four panels, the weights are computed by the two leading diffusion vectors of the graphs G_A (left two panels) and G_B (right two panels). In the bottom four panels, the weights are equal to the differential vectors V_A (left two panels) and V_B (right two panels).

Clearly, the differential vectors highlight genes that are more relevant for differentiating between the two groups.

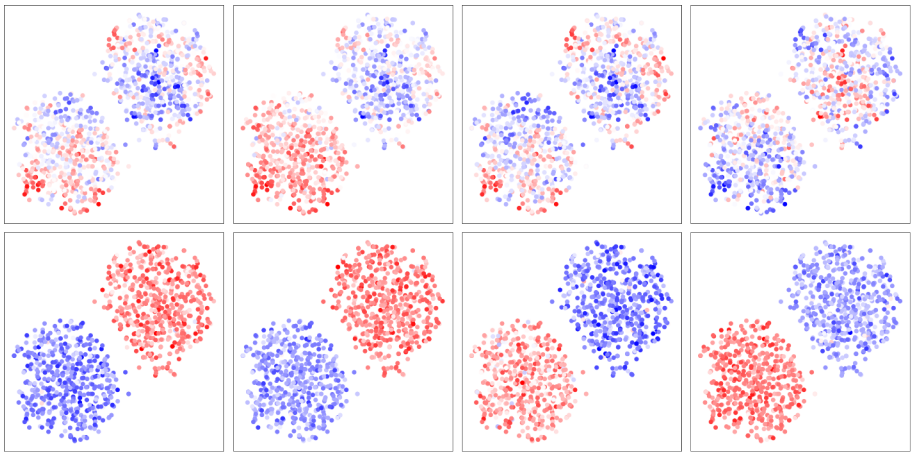


Figure 2.6. 2D t-SNE plots of simulated scRNA-seq generated by splatter Zappia et al. [2017]. The two clusters represent two cell types. In all panels cells are colored by *meta-features* - a weighted sum on the gene expression, corresponding to diffusion maps on each of the feature graphs separately (top row), or DiSC vectors (bottom row).

2.4.4 Differential pixel structures in handwritten digit images

The MNIST dataset [LeCun and Cortes, 2010] consists of images of hand written digits from 0 to 9, each of dimension 28×28 pixels. It has 60k training samples and 10k testing samples. We used Algorithm 2.2, DiSC on multiple datasets, on the MNIST dataset to extract differential features for each of the digits: features present in a single digit but are not present in each of the other digits. To have a quantitative metric to measure the significance of these differential features, we develop a classifier for a 10-class classification problem. For this, we use K-means clustering on the differential features for each digit (K=10) thus deriving clusters of pixels that together differentiate between digits. We then compute a meta-feature by averaging pixel values for each cluster. Logistic regression is performed on these meta features to distinguish between the 10 classes.

We compare our approach with Diffusion maps, Elastic Net (EN) and Elastic Net - logistic. For diffusion maps, we replaced the differential features with diffusion maps and followed the

same procedure. For EN and EN-logistic, we obtained the feature importance vectors by training a classifier to distinguish between the 10-classes. K-means clustering is performed on these feature importance vectors. This cluster information is further used to compute meta-feature and finally for classification. Additionally, the entire data (784 features) is used to obtain the clusters using K-means clustering which are further used to compute classification accuracy following the procedure mentioned above. We consider this as a baseline. The results are tabulated below in Table 2.1.

We can see that our proposed method has the best performance, as it can capture the important differential information needed for classification. Since diffusion maps capture both shared and differential features, and entire data has all the information, their performance is poor compared to DiSC. Elastic Net has the least accuracy because it is designed for a regression setup but this is a classification problem. Elastic Net-logistic has better performance over Elastic Net as we use a cross entropy loss function – the one used for classification setup. Further, DiSC has better performance than Entire Data as it is designed to capture the differential features. However, DiSC outperformed EN-logistic as it is designed to extract groups of differential features unlike EN-logistic which just extracts all the differential features as a single group, especially when those features have similar effect on the classification problem.

Table 2.1. Classification performance on MNIST using DiSC and other competing approaches.

Method	Test Accuracy (%)
Entire Data	76.23
Diffusion Maps	87.42
Elastic Net	69.96
Elastic Net-logistic	82.42
DiSC (Ours)	**89.75**

2.4.5 Change detection in hyperspectral imagery

We apply DiSC to a hyperspectral imagery change detection dataset [Eismann et al., 2007], consisting of hyperspectral images of a fixed outdoor scene captured under varying

weather conditions, illumination, and across four different months of the year (August, September, October, and November). Each hyperspectral image has a spatial resolution of 100×100 pixels with 124 spectral bands. We denote these datasets by $X^{\text{aug}}, X^{\text{sep}}, X^{\text{oct}}$, and X^{nov}. The scene contains a metal frame, grass, and trees in the background, as shown in Fig. 2.7(a).

An additional hyperspectral image, referred to as *October-change* and denoted $X^{\text{oct-c}}$, was acquired in October after introducing a new object—a tarp—into the scene (Fig. 2.7(a)). The objective is to identify the tarp as an added object while being robust to other variations induced by acquisition conditions such as lighting and weather.

We consider $X^A \in \{X^{\text{aug}}, X^{\text{sep}}, X^{\text{oct}}, X^{\text{nov}}\}$ and $X^B = X^{\text{oct-c}}$, and compute differential features for each month-to-change pair. This setting mirrors the toy problem introduced in Sec. 2.4.1 and the theoretical analysis in Sec. 2.3.1, where a group of correlated features in X^A becomes subdivided in X^B. Here, pixels corresponding to grass form a correlated group in X^A, while in X^B this group is split into grass and tarp pixels. The tarp pixels remain mutually correlated but lose connectivity with the surrounding grass, creating a distinct differential structure.

For each pair, we extract the top four significant differential vectors of X^B and apply k-means clustering ($k = 3$) in this reduced space. Figure 2.7(b) shows the resulting feature clusters for all four month-to-change comparisons. In all cases, the tarp is consistently identified as a dominant and coherent cluster of differential features, despite substantial variability in acquisition conditions across months. We compare these results with Diffusion Maps for changing data (DM-changing data) [Coifman and Hirn, 2014], a spectral approach designed to capture differences between evolving datasets by defining a distance between diffusion embeddings. Pixel-wise distances between X^A and X^B are clustered into three groups, as shown in Fig. 2.7(c). In contrast to DiSC, DM-changing data is highly sensitive to global acquisition differences and often groups additional background regions together with the tarp, particularly under varying lighting or weather conditions.

To further illustrate the feature grouping capability of DiSC, we focus on the October

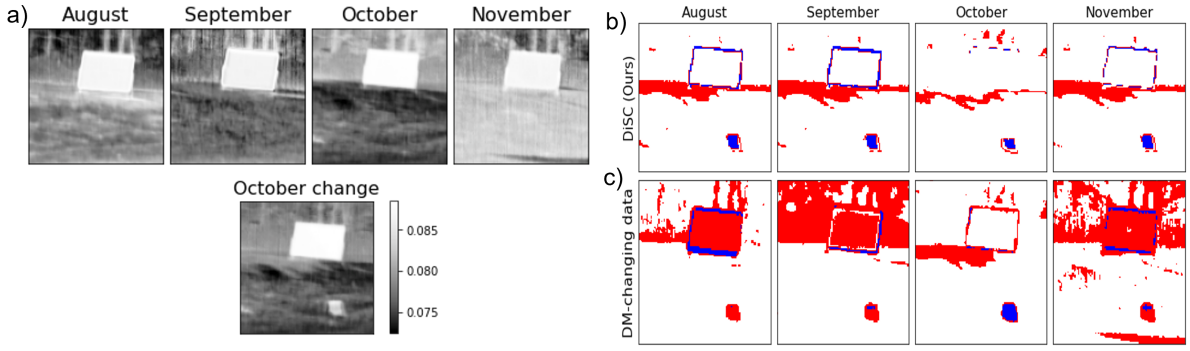


Figure 2.7. Hyperspectral imaging averaged across channels (a). Three clusters of features (white, red and blue) formed from DiSC (b) and Diffusion Maps for changing data (c).

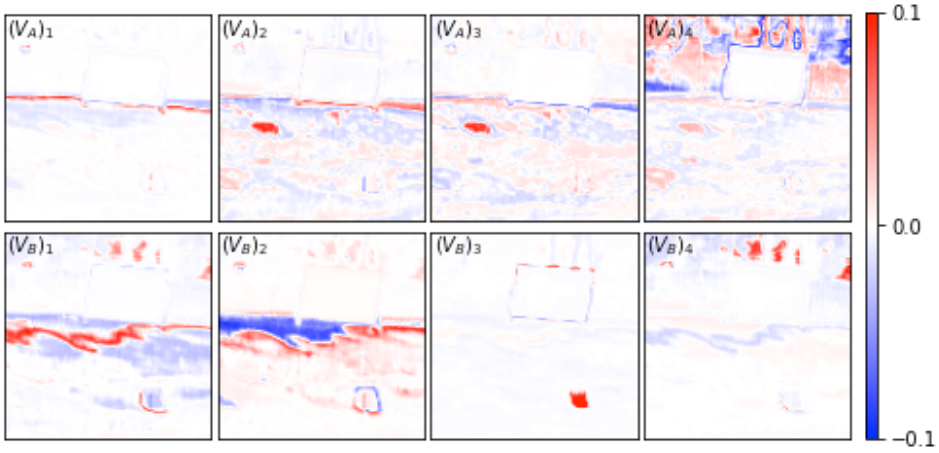


Figure 2.8. The differential vectors for $X^A = X^{oct}$ (top row) and $X^B = X^{oct-c}$ (bottom row).

pair ($X^A = X^{\text{oct}}$, $X^B = X^{\text{oct-c}}$) and set $d_A = d_B = 20$. The top four differential vectors for X^A and X^B are shown in Fig. 2.8. The vectors $(V_B)_3$ and $(V_B)_4$ isolate the tarp and subtle variations in the background trees, respectively, while $(V_B)_1$ and $(V_B)_2$ capture changes within the grass region. Each differential structure is recovered as a separate vector, demonstrating DiSC's ability to disentangle multiple sources of change. Since all structures present in X^A persist in X^B, the differential vectors of X^A do not capture meaningful differences.

Finally, we evaluate robustness to noise by adding Gaussian noise with mean $\mu = 0$ and standard deviation $\sigma = 0.01$ to all datasets. Differential features are recomputed using DiSC and compared with DM-changing data, followed by k-means clustering ($k = 3$). Figure 2.9 shows

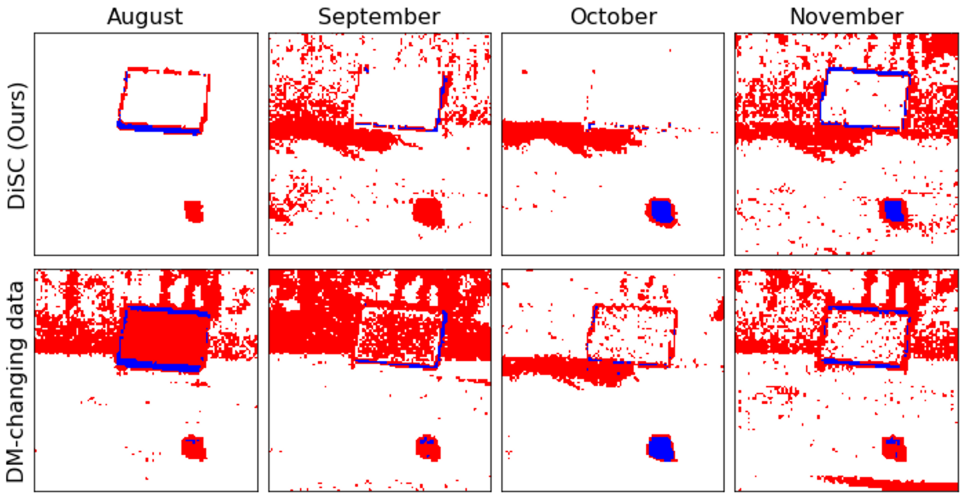


Figure 2.9. Noisy data. Three groups of features (white, red and blue) formed from DiSC (top row) and Diffusion Maps for changing data (bottom row).

that DiSC consistently isolates the tarp as a distinct cluster across months, whereas DM-changing data frequently merges the tarp with unrelated background regions. These results demonstrate that DiSC is more robust to noise and acquisition variability, highlighting object-level changes while remaining invariant to global imaging conditions.

2.4.6 Hyperparameter selection

The Disc algorithm requires the following hyperparameters: (i) The bandwidth for the kernel functions $K_A(X^A_{\cdot i}, X^A_{\cdot j}), K_B(X^B_{\cdot i}, X^B_{\cdot j})$, see Eq. (2.1), and (ii) The number of significant eigenvectors computed for G_A and G_B, denoted d_A and d_B, respectively.

Self-tuning bandwidth

For computing the weight matrices, we use the self-tuning bandwidth from Zelnik-Manor and Perona [2004] where the bandwidth for an RBF kernel is given by $K(x_i, x_j) = \exp(\|x_i - x_j\|/\sigma_i\sigma_j)$. The local bandwidth σ_i for each node is set to the distance to its k-th nearest neighbor, as suggested in Zelnik-Manor and Perona [2004] and as is common in practice. The rule of thumb for choosing k is around $\log(p)$ where p is the number of features.

Determining d_A and d_B

The notation d_A, d_B represent the number of significant eigenvectors present in the random walk matrices of P_A, P_B respectively. It is important to note that if d_A, d_B are two small, the leading singular vectors of $P_A Q_B$ and $P_B Q_A$ will include elements of the shared latent space. The results will not change dramatically, however, if the choice of d_A, d_B is higher than the optimum. For example, in the experiment in section 4.1 "Identifying newly connected features", when we consider $d_A = d_B < 4$, the shared latent space of features between 100-150 are also highlighted in V_A, V_B which is undesirable. However, we can increase the values of d_A, d_B up to around 150, with very little impact on the results. For the MNIST data, we computed differential features between digits 4 and 9 with various values for $d_A = d_B = d$ and followed the procedure mentioned in the paper to compute the classification accuracy. These results are given in Table 2.2. We can see that for very small values of d , the accuracy is lower since the classifier is partially trained on information about shared features. Here as well, there is a wide range of values (between 20-40) that yield similar results.

Table 2.2. Impact of the choice of hyperparameter d on the classifier accuracy for pairs of MNIST digits.

$d_A = d_B = d$	Test Accuracy (%)
10	95.2
20	96.5
30	96.5
40	96.4
50	94.7
60	88.5

Acknowledgements

Chapter 2, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Mishne, Gal; Jaffe, Ariel (2022). "DiSC: Differential Spectral Clustering of Features." In *Advances in Neural Information Processing Systems 35 (NeurIPS)*. The dissertation author is the primary author of this paper.

Chapter 3

Conditional Stochastic Gates for Contextual Feature Selection

3.1 Motivation for contextual feature selection

Feature selection techniques are central to machine learning and data analysis, as they aim to identify informative variables from high-dimensional observations while discarding nuisance or redundant features. This problem has become increasingly important across scientific domains due to the rapid growth of data dimensionality and the complexity of modern predictive models. By isolating informative features, feature selection can simplify models, improve generalization, and enhance interpretability [Li et al., 2017, Kumar and Minz, 2014, Islam et al., 2022, Everaert et al., 2022]. Broadly, existing feature selection approaches are commonly categorized into filter, wrapper, and embedded methods.

Filter methods [Battiti, 1994, Peng et al., 2005, Estévez et al., 2009, Song et al., 2007, 2012, Chen et al., 2017, Sristi et al., 2022] rely on predefined criteria that are independent of the predictive model, ranking features based on statistical measures such as correlation, mutual information, or dependence scores [Lewis, 1992]. Wrapper methods [Kohavi and John, 1997, Stein et al., 2005, Zhu et al., 2007, Reunanen, 2003, Allen, 2013] evaluate subsets of features by explicitly training and testing a predictive model, selecting the subset that yields the best performance. While effective in some settings, wrapper methods are often computationally prohibitive for high-dimensional data or complex models. Embedded methods [Tibshirani,

1996a, Hans, 2009, Li et al., 2011, 2006] incorporate feature selection directly into the learning process, typically through regularization mechanisms. Prominent examples include ℓ_1-based approaches such as LASSO and its variants [Daubechies et al., 2010, Bertsimas et al., 2017], which promote sparsity in model parameters. Embedded methods are particularly appealing in scientific applications, as they provide a unified framework for learning predictive models and identifying informative features simultaneously.

Most existing feature selection techniques are *global* in nature, in the sense that they identify a single set of informative features assumed to be relevant across all samples and experimental conditions. However, in many real-world settings—particularly in neuroscience, healthcare, and behavioral sciences—the relevance of features depends strongly on contextual variables. Context may include task phase, stimulus configuration, demographic variables, time, or experimental parameters. Importantly, such contextual variables do not necessarily encode the outcome of interest on their own, but instead modulate how information is represented and which features are informative under a given set of circumstances.

In neuroscience, this distinction is particularly critical. Neural populations often exhibit context-dependent representations, where the same neurons can encode information differently depending on task demands, sensory conditions, or internal state. Assuming a fixed, global notion of feature importance can obscure this structure: neurons that are informative only in specific contexts may appear weak or irrelevant when averaged across all conditions. As a result, global feature selection methods may conflate distinct representational regimes, limiting interpretability and masking how neural systems adapt their encoding strategies.

A common strategy for incorporating context is to concatenate contextual variables with explanatory features and train a single predictive model. While straightforward, this approach does not explicitly reveal how feature relevance depends on context and often leads to models that are difficult to interpret. Another alternative is to train separate models for different values of categorical contextual variables (e.g., separate models for different task conditions or demographic groups). However, this approach scales poorly as the number of contextual variables

increases, reduces the effective sample size available to each model, and becomes impractical for continuous contextual variables, which require arbitrary discretization. Such binning can obscure smooth or systematic changes in feature relevance and further limit interpretability, especially in the presence of multiple contextual variables.

Several methods have been proposed to address context-specific feature selection more explicitly. Contextual Explanation Networks (CEN) [Al-Shedivat et al., 2020], Contextualized Machine Learning [Lengerich et al., 2023], and Contextual LASSO [Thompson et al., 2023] model the dependence of predictive parameters on contextual variables, typically within a linear modeling framework. While these approaches provide valuable insights, they often lack explicit sparsity constraints or rely on ℓ_1 regularization, which can lead to coefficient shrinkage and reduced interpretability. Moreover, many of these methods are limited to linear prediction models, restricting their applicability in settings where nonlinear relationships are essential.

More recently, dynamic and sample-wise feature selection methods have been introduced. Sample-specific approaches [Chen et al., 2018, Yoon et al., 2018, Yang et al., 2022a] tailor feature selection to individual samples based on their feature values, requiring all features to be observed in advance. Active feature acquisition methods [Shim et al., 2018, Covert et al., 2023] select features sequentially during inference. While both paradigms can, in principle, incorporate contextual information—often by concatenation or prioritization of contextual variables—the resulting interpretability is limited. Understanding how feature importance varies as a function of context typically requires aggregating selections across samples and resorting to coarse binning, which again obscures smooth contextual structure.

These limitations motivate the need for feature selection frameworks that explicitly model how relevance varies with context, while remaining interpretable, scalable, and applicable to both linear and nonlinear prediction models. From a population-level perspective, such methods should reveal how the importance of individual features—or neurons—changes systematically with contextual variables, rather than assuming a fixed mapping between features and outcomes. By treating context as a modulator of representation rather than as a direct predictive target,

this perspective provides a principled foundation for studying adaptive, context-dependent representations in neural populations and beyond.

3.2 Problem setup and background

Let $X \subseteq \mathbb{R}^D$, $Z \subseteq \mathbb{R}^L$ and $Y \subseteq \mathbb{R}$ be the random variables corresponding to the input explanatory features, input contextual variables, and output prediction value, respectively. Given the realizations of these random variables from some unknown joint data distribution $P_{X,Z,Y}$, the goal of embedded contextual feature selection methods is to achieve the following simultaneously: 1) construct a hyper-model $h_{\boldsymbol{\phi}} : \mathbb{R}^L \to \{0,1\}^D$ that selects a subset of explanatory features $X_{\mathscr{S}}$ as a function of context z; and 2) construct a model $f_{\boldsymbol{\theta}} : \mathbb{R}^D \to \mathbb{R}$ that predicts the response Y based on these selected features. Given a loss function L, we solve for the parameters θ and ϕ of the risk minimization problem

$$R(\theta, \phi) = \mathbb{E}_{X,Z,Y}[L(f_\theta(x \odot s(z)), y)], \tag{3.1}$$

where $s(z) = h_{\boldsymbol{\phi}}(z)$ and x, z and y represent a realization of the random variables X, Z and Y following a data distribution $P_{X,Z,Y}$, the feature selection vector, given by a vector of indicator variables for the set $\mathscr{S}$, is the output of the hypernetwork $s(z) = h_{\boldsymbol{\phi}}(z) = \{0,1\}^D$, and $\odot$ denotes the point-wise product.

Throughout this paper, bold lowercase letters denote vectors, e.g., x, while scalars are represented by unbolded lowercase letters. Elements within a vector are indicated by subscripts, such as $x_1, x_2, \ldots, x_n$. For a function f that maps a vector x to another vector, $f(x)$, the i^{th} element of the resultant vector is denoted as $f_i(x)$. 0 and 1 indicate a vector of all zeros and all ones, respectively. I represents identity matrix. $d \in [D]$ is shorthand for d ranging from 1 to D, inclusive.

3.3 Conditional stochastic gates

The risk minimization in (3.1) often includes an additional constraint to induce sparsity on the feature selection, for example, $\|s\|_0 \leq D$, which reduces the number of selected features and enhances interpretability. Model interpretability is crucial for understanding complex relations between input features and the predicted target in applications such as health care [Liu et al., 2019], shopping [Baltrunas and Ricci, 2009], and psychology [Everaert et al., 2022].

In practice, ℓ_0 regularization is computationally challenging, especially for high dimensional data, and cannot be integrated into gradient descent-based optimization commonly used in deep networks. Our proposed solution is a probabilistic and computationally efficient method that facilitates contextual feature selection. This is achieved by applying ℓ_0 regularization to stochastic gates, which mask the input feature conditioned on the context.

First, we introduce conditional Bernoulli gates corresponding to each of the D features as a probabilistic extension of contextual feature selection. Expressly, we assume that the probability of an individual explanatory feature being selected, given contextual features z, follows a Bernoulli distribution independent of the probability of selecting other explanatory features. Let $S'|Z$ be a conditional random vector that represents these independent Bernoulli gates, with $\mathbb{P}(S'_d = 1|Z = z) = \pi_d(z)$ for $d \in [D]$, and let $s'(z) = s'|z$ be a realization of the random vector $S'|Z$. We term these conditional-Stochastic Gates (c-STG), as the parameter of the distribution of each gate is conditional on the context variables z.

The hypernetwork $h'_{\boldsymbol{\phi}}$ learns a parametric function mapping the contextual variable z to the conditional probability, i.e., the parameter of the Bernoulli distribution. The task of contextual variable selection then boils down to learning the parameters $\pi(z) = h'_{\boldsymbol{\phi}}(z)$ of this conditional distribution, which spans over a continuous space $[0,1]^D$, instead of a discrete set $\{0,1\}^D$, as in Eq. (3.1). Let $h'_{\boldsymbol{\phi}} : \mathbb{R}^L \to [0,1]^D$ be the function that maps the contextual information to the parameters of the probability distribution. This reformulates the regularized version of the risk in Eq. (3.1) to

$$\hat{R}(\theta,\phi) = \hat{\mathbb{E}}_{X,Z,Y}\mathbb{E}_{S'|Z}[L(f_{\theta}(x \odot s'(z)), y) + \lambda ||s'(z)||_0], \tag{3.2}$$

where the parameters of the distribution $S'|Z$, $\pi(z)$, are learnt by the hypernetwork, $h'_{\phi}(z)$. $\hat{\mathbb{E}}_{X,Z,Y}$ represents the empirical expectation over the observations X, Z and Y and $\mathbb{E}_{S'|Z}||s'(z)||_0 = \sum_{d=1}^{D} \pi_d(z)$, the sum of Bernoulli parameters. Constraining $\pi_d(z)$ to $\{0,1\}$ makes this equivalent to the cardinality-constrained version of Eq. (3.1) , with a regularized penalty on cardinality rather than an explicit constraint. Moreover, this probabilistic formulation converts the combinatorial search to a search over the space of Bernoulli distribution parameters. Thus, the problem of feature selection translates to finding θ^* and ϕ^* that minimize the empirical risk based on the formulation in Eq. (3.2).

Incorporating discrete random variables into a differentiable loss function to retain informative data features is appealing, but discrete variable gradient estimates often have high variance He and Niyogi [2004]. Consequently, continuous approximations of discrete variables have been proposed Maddison et al. [2017], Jang et al. [2017]. A stable approach for continuous relaxation uses the Gaussian distribution, more consistent in feature selection than Gumbel-softmax techniques like concrete and hard concrete Jang et al. [2017], which can lead to high variance in approximating Bernoulli variables Yamada et al. [2020], Jana et al. [2023]. Such relaxations Louizos et al. [2018] are applied in various areas, including discrete softmax activations Jang et al. [2017], feature selection Yamada et al. [2020], Lindenbaum et al. [2021b], Shaham et al. [2022], and sparsification Lindenbaum et al. [2021a, 2024]. We utilize a Gaussian-based relaxation for Bernoulli variables, termed Stochastic Gates (STG) Yamada et al. [2020], differentiated using the reparameterization trick Miller et al. [2017], Figurnov et al. [2018].

The Gaussian-based continuous relaxation for the Bernoulli variable is defined as $\widetilde{s}_d(z) = \max(0, \min(1, \mu_d(z) + \varepsilon_d)$, where ε_d is drawn from a normal distribution $\mathcal{N}(0, \sigma^2)$, with σ fixed throughout training. Unlike the ReLU function, which only clips the negative values to zero, the

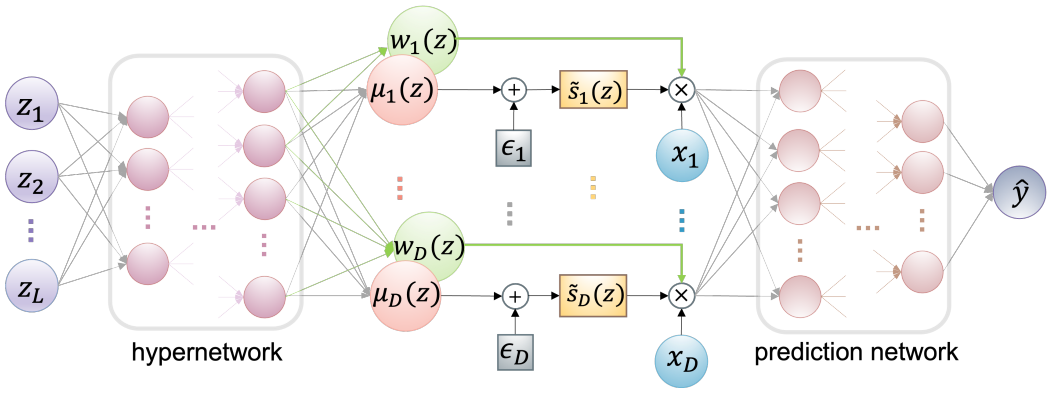


Figure 3.1. Contextual feature selection framework. Contextual variables z (in purple) feed into the hypernetwork. The hypernetwork outputs the parameters of the gates, $\mu(z)$, which are combined with ϵ to determine if each gate is open or close $\tilde{s}_d$ (yellow) for each feature x_d (blue). For weighted c-STG, the hypernetwork also outputs weight vectors (green), indicating the importance of the selected explanatory features. The selected and weighted features are fed into the prediction model, thus enhancing its ability to process feature significance in predictions.

mean-shifted Gaussian variable clips both the positive and negative values; therefore, it accounts for the binary nature of the original random variable. Here, we learn μ_d as a parametric function of the contextual variables z. Thus, the hypernetwork $\widetilde{h}_{\boldsymbol{\phi}}$ aims to learn the parameters of the relaxed-continuous distribution as a function of the context variables z instead of learning the original discrete distribution. Our objective as a minimization of the empirical risk $\hat{R}(\boldsymbol{\theta}, \boldsymbol{\phi})$ is as follows:

$$\min_{\boldsymbol{\theta},\boldsymbol{\phi}} \hat{\mathbb{E}}_{X,Z,Y} \mathbb{E}_{\widetilde{S}|Z} \left[L(f_{\boldsymbol{\theta}}(x \odot \widetilde{s}(z)), y) + \lambda ||\widetilde{s}(z)||_0 \right], \tag{3.3}$$

where $\widetilde{S}|Z$ is a random vector with D independent variables $\widetilde{s}_d(z) = \widetilde{s}|z$ for $d \in [D]$ and the parameters of the distribution $\widetilde{S}|Z$, $\pi(z)$, are learnt by the hypernetwork, $\widetilde{h}_{\boldsymbol{\phi}}(z)$. The regularization term can be further simplified to

$$\mathbb{E}_{\widetilde{S}|Z} ||\widetilde{s}(z)||_0 = \sum_{i=1}^{D} \mathbb{P}(\widetilde{s}_d(z) > 0) = \sum_{d=1}^{D} \Phi(\frac{\mu_d(z)}{\sigma}), \tag{3.4}$$

where Φ is the standard Gaussian CDF. The term (3.4) penalizes open gates so that gates corresponding to features that are not useful for prediction are encouraged to transition into a closed state (which is the case for small $\mu_d(z)$). Hence, we perform a context-specific feature selection strategy by inducing sparsity through the empirical mean of the regularization term (3.4) over multiple realizations of Z. This enables us to select distinct informative features for different contexts while maintaining the sparsity.

In practice, we consider the function class for $\widetilde{h}_{\boldsymbol{\phi}}$ and $f_{\boldsymbol{\theta}}$ to be a class of neural networks parameterized by $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$, respectively. To optimize for these parameters, we use a Monte Carlo sampling gradient estimator of (3.3), which gives

$$\begin{aligned}
\frac{\partial \hat{R}(\boldsymbol{\theta},\boldsymbol{\phi})}{\partial \boldsymbol{\theta}} &= \frac{1}{K}\sum_{k=1}^{K}\frac{\partial}{\partial \boldsymbol{\theta}} L(f_{\boldsymbol{\theta}}(x^{(k)} \odot \widetilde{s}(z^{(k)}), y^{(k)}) \\
\frac{\partial \hat{R}(\boldsymbol{\theta},\boldsymbol{\phi})}{\partial \boldsymbol{\phi}} &= \frac{1}{K}\sum_{k=1}^{K}\frac{\partial}{\partial \phi} \mathscr{L}(f_{\theta}(x^{(k)} \odot \widetilde{s}(z^{(k)})), y^{(k)}) + \lambda \frac{\partial}{\partial \phi}\Phi\left(\frac{\widetilde{h}_{\phi}(z^{(k)})}{\sigma}\right)
\end{aligned} \tag{3.5}$$

where K is the number of Monte Carlo samples (corresponds to the batch size). Our methodology is summarized in Alg. 3.1 and illustrated in Fig. 3.1. Note for c-STG, the explanatory features are masked by the feature gates, $\widetilde{s}(z)$, and fed into the prediction model $\hat{y}^{(k)} = f_{\boldsymbol{\theta}}(x^{(k)} \odot \widetilde{s}(z^{(k)}))$. In the initial training phase $\boldsymbol{\phi}$, all gates should have an equal probability of being open or closed. We set $\mu_d(z) = 0.5 \ \forall \ d \in [D]$ so that all gates approximate a "fair" Bernoulli variable. The initialization of the hyper-model's $\boldsymbol{\phi}$ using Xavier initialization Glorot and Bengio [2010] and a Sigmoid activation function in the final layer of $\widetilde{h}_{\boldsymbol{\phi}}$ ensures that the means ($\mu_d(z)$) are centered around 0.5, $\forall\ z$, early in the training phase. It is worth noting that we need the noise term only during the training phase.

3.4 Theoretical analysis

We conduct a thorough theoretical analysis to establish the equivalence between our probabilistic formulation, as represented by Eq.3.2, and the original NP-hard contextual variable

selection problem defined in Eq.3.1. Additionally, we provide proof demonstrating that c-STG achieves a lower risk than STG. Furthermore, we extend our analysis to a linear regression scenario, where we demonstrate the main advantage of c-STG over STG. While STG selects features with consistent significance across contextual variables on average, c-STG adapts the feature selection process according to the contextual variables, thus effectively learning the optimal feature selection as a function of these variables.

Theorem 3.1. *Let $s^*(z)$ and $s'^*(z)$ represent the optimal feature selection functions in Eq.* (3.1) *and its corresponding probabilistic formulation in Eq.* (3.2) *respectively. Then $s^*(z) = s'^*(z)$.*

Proof: Firstly, the optimal solution for Eq. (3.1) lies in the feasible solution space for Eq. (3.2). Secondly, Eq. (3.2) attains its optimal value at the extreme values of the Bernoulli parameters $\pi(z)$. We refer the reader to Yin et al. [2022] for detailed proof.

This theorem suggests that a deterministic search for feature selection can be transitioned to a probabilistic approach. We use the universal function approximators, deep networks, to learn the function $s'^*(z)$. In subsequent theorems, we contrast c-STG's and STG's performance and feature selection abilities. In Yamada et al. [2020], the optimization problem of the global feature selection using STG is given by

$$\min \hat{\mathbb{E}}_{X,Y} \mathbb{E}_{S'} \left[L(f_\theta(x \odot s'), y) + \lambda ||s'||_0 \right], \tag{3.6}$$

where S' is a random vector that represents independent Bernoulli gates, $\mathbb{P}(S'_d = 1) = \pi_d$ for $d \in [D]$, and s'_d denotes the realization of the random vector S'_d. Note that S' is similar to $S'(z)$ except that the latter is a function of z, and the former is constant for all z. Thus, their approach optimizes for a fixed feature selection independent of contextual information by promoting sparsity in feature selection through the regularization term $\mathbb{E}_{\widetilde{S}}[||\widetilde{s}||_0]$. In contrast, we perform context-specific feature selection by maximizing the sparsity through the empirical mean of Eq. (3.4) across various realizations of Z. The following theorem draws a connection between their empirical risk minimizations.

Theorem 3.2. *c-STG attains an optimal risk lower or equal to the risk attained by STG.*

Proof: The feasible solution space of STG is contained in the feasible solution space of c-STG as the parameters of the stochastic gates in c-STG can be any function of contextual variables z, which includes a constant function, which is the case in STG. Therefore, the optimal risk attained by c-STG must be less than or equal to the optimal risk value attained by STG.

Through the following Theorems 3.3 and 3.4, we further emphasize the advantage of c-STG over STG.

Theorem 3.3. *In a linear regression setup, the relationship between the optimal parameters of the conditional Bernoulli stochastic gates, $\pi^*(z)$, and the optimal parameters of the non-conditional Bernoulli stochastic gates π^*_{stg} is given by*

$$\pi^*_{stg} = E_Z[\pi^*(z)]. \tag{3.7}$$

Proof: The proof proceeds in five steps.

Step 1: When we consider $L(\cdot,\cdot)$ to be a mean squared loss, the solution to the optimization problem in Eq. (3.6) and Eq. (3.2) will be given by their respective MMSE (minimum mean squared error) estimates. These are given by,

$$f_{\boldsymbol{\theta}^*}(x \odot s') = E[Y|X = x, S' = s'] \tag{3.8}$$

$$f_{\boldsymbol{\theta}^*}(x \odot s'(z)) = E[Y|X = x, Z = z, S'(z) = s'(z)] \tag{3.9}$$

where the feature selection vector s' and contextual feature selection vector $s'(z)$ will be sampled from a constrained space.

Step 2: Relaxing $f_{\boldsymbol{\theta}}$ to be a linear function $f_{\boldsymbol{\theta}}(x) = \theta^T x$, the above MMSE estimates are further reduced to

$$
\begin{aligned}
E[Y|X=x,S'=s'] &= f_{\boldsymbol{\theta}^*}(x\odot s') \\
&= \theta^{*T}(x\odot s') \\
&= (x\odot\theta^*)^T s'
\end{aligned}
\tag{3.10}
$$

$$
\begin{aligned}
E[Y|X=x,Z=z,S'(z)=s'(z)] &= f_{\boldsymbol{\theta}^*}(x\odot s'(z)) \\
&= \theta^{*T}(x\odot s'(z)) \\
&= (x\odot\theta^*)^T s'(z)
\end{aligned}
\tag{3.11}
$$

Step 3: We now average the estimates given in Eq. (3.10) and Eq. (3.11) across S' and $S'|Z=z$ respectively.

$$
E[Y|X=x] = E_{S'}(x\odot\theta^*)^T s' = (x\odot\theta^*)^T\pi^*_{\text{stg}} \tag{3.12}
$$

$$
\begin{aligned}
E[Y|X=x,Z=z] &= E_{S'(z)}(x\odot\theta^*)^T s'(z) \\
&= (x\odot\theta^*)^T\pi^*(z)
\end{aligned}
\tag{3.13}
$$

Step 4: We now project the MMSE estimate of c-STG onto the space of contextual-feature selection vectors that remain constant across contextual variables. This is given by

$$
\begin{aligned}
E_Z E[Y|X=x,Z=z] &= E_Z(x\odot\theta^*)^T\pi(z) \\
&= (x\odot\theta^*)^T E_Z\pi^*(z)
\end{aligned}
\tag{3.14}
$$

Step 5: This solution should be equivalent to solving the STG problem as it restricts the stochastic gates to be constant over z. Comparing the two equations Eq. (3.12) and Eq. (3.14), we get $\pi^*_{\text{stg}} = E_Z[\pi^*(z)]$.

Theorem 3.4. *In a linear regression setup, the relationship between the optimal parameters of the conditional Gaussian stochastic gates, $\mu^*(z)$, and the optimal parameters of the non-conditional Gaussian stochastic gates μ^*_{stg} is given by*

$$
\mu^*_{stg} = E_Z[\mu^*(z)]. \tag{3.15}
$$

Proof: This follows from the proof of Theorem 3.3 by replacing Eq. (3.12) and Eq. (3.13) with the following two equations respectively:

$$E[Y|X=x] = E_{S'}(x \odot \theta^*)^T s' = (x \odot \theta^*)^T \mu^*_{\text{stg}} \tag{3.16}$$

$$\begin{aligned} E[Y|X=x, Z=z] &= E_{S'(z)}(x \odot \theta^*)^T s'(z) \\ &= (x \odot \theta^*)^T \mu^*(z) \end{aligned} \tag{3.17}$$

The theorems illustrate that c-STG offers an enhanced feature selection resolution by pinpointing features crucial to specific contexts. This stands in contrast to STG, which tends to select features based on their average importance across various contexts. Theorem 3.3 delineates this difference in the realm of discrete probability spaces, whereas Theorem 3.4 addresses the continuous probability spaces.

3.5 Weighted conditional stochastic gates

We extend c-STG to weighted conditional-STG (weighted c-STG) to determine and quantify the significance of the features identified by the conditional stochastic gates. To achieve this, we integrate an additional layer in our model that maps the hypernetwork's penultimate layer, $h_L(z)$, to a weight vector $w(z) = Wh_L(z) + b$, as depicted in Figure 3.1 (green circles). Explanatory features x are masked by the feature selection output $\widetilde{s}(z)$ and weighted by $w(z)$, then fed into the prediction model for task execution. The hypernetwork and prediction network parameters are learned using back propagation, as detailed in Algorithm 3.1.

Theorem 3.5. *Weighted c-STG attains an optimal risk lower or equal to the risk attained by c-STG.*

Proof: This follows similar to the proofs of the theorem 3.2. The feasible solution space of c-STG is contained in the feasible solution space of weighted c-STG as the weights of the weighted c-STG can be any function of contextual variables z, which includes a constant function,

Algorithm 3.1. Weighted c-STG

Input: $x^{(k)} \in \mathbb{R}^D, z^{(k)} \in \mathbb{R}^L, y^{(k)} \in \mathbb{R}$, for $k = [K]$
Output: Trained models $f_{\boldsymbol{\theta}}$ and $\tilde{h}_{\boldsymbol{\phi}}$.

Initialize: θ and $\boldsymbol{\phi}$ using Xavier initialization.
while model not converged **do**
    Forward Pass:
    for $k = 1$ to K **do**
        $\mu(z^{(k)}), w(z^{(k)}) = \widetilde{h}_{\boldsymbol{\phi}}(z^{(k)})$
        $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$
        $\widetilde{s}(z^{(k)}) = \max(0, \min(1, \mu(z^{(k)}) + \varepsilon))$.
        where min and max are applied elementwise.
        $\hat{y}^{(k)} = f_{\boldsymbol{\theta}}(x^{(k)} \odot \widetilde{s}(z^{(k)}) \odot w(z^{(k)}))$
    end for
    $\hat{R}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \sum_{k=1}^{K} \left[L(\hat{y}^{(k)}, y^{(k)}) + \lambda \sum_{d=1}^{D} \Phi\left(\frac{\mu_d(\boldsymbol{z}^{(k)})}{\sigma}\right) \right]$
    Back Propagation:
    Update θ: $\theta \leftarrow \theta - \eta \frac{\partial \hat{R}(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\theta}}$
    Update ϕ: $\phi \leftarrow \phi - \eta \frac{\partial \hat{R}(\boldsymbol{\theta}, \boldsymbol{\phi})}{\partial \boldsymbol{\phi}}$
end while

which is the case in c-STG. Therefore, the optimal risk attained by weighted c-STG must be less than or equal to the optimal risk value attained by c-STG.

3.6 Empirical evaluation and applications

3.6.1 Controlled toy problems

We first validate c-STG on synthetic datasets where the set of informative features varies deterministically with context. These controlled experiments allow us to assess whether c-STG can recover context-specific feature relevance under known ground-truth structure, and to isolate failure modes of existing context-aware and population-level baselines.

Following prior work Yamada et al. [2020], Yang et al. [2022b], we construct a moving

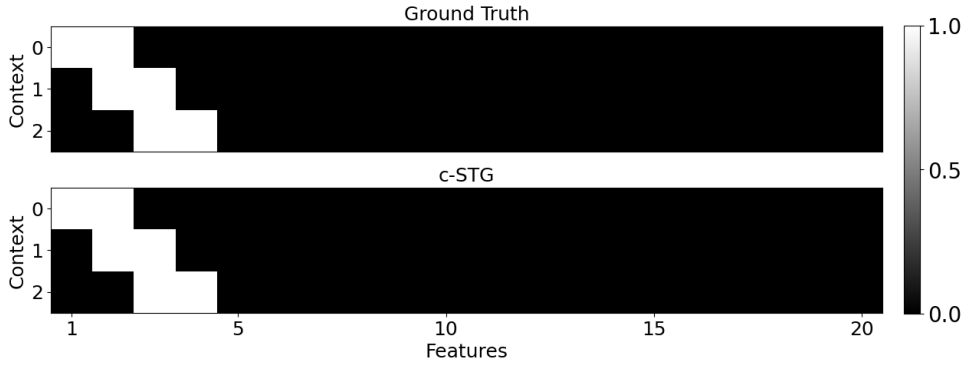


Figure 3.2. XOR1. Ground-truth feature relevance (top) and features identified by c-STG (bottom), demonstrating accurate recovery of context-specific dependencies.

XOR dataset in which different feature pairs determine the response depending on context:

$$\textbf{XOR1:} \quad y(\mathbf{x},\mathbf{z}) = \begin{cases} x_1 x_2, & z = 0, \\ x_2 x_3, & z = 1, \\ x_3 x_4, & z = 2. \end{cases} \tag{3.18}$$

Figure 3.2 compares the ground-truth context-dependent feature relevance with the features selected by c-STG. c-STG accurately recovers the correct feature pair for each context, assigning high gate values exclusively to the relevant features while suppressing irrelevant ones. This qualitative agreement with ground truth is reflected quantitatively in Table 3.1, where c-STG achieves perfect prediction accuracy on XOR1. While several baseline methods also attain non-trivial accuracy on this task, Table 3.2 shows that they do so by selecting a substantially larger number of features. In contrast, c-STG achieves perfect prediction using the minimal number of context-relevant features, demonstrating that it recovers the correct conditional structure rather than relying on spurious or redundant features. Together, these results highlight that the advantage of c-STG lies not merely in predictive performance, but in its ability to recover interpretable, context-specific feature relevance.

We next consider a nonlinear weighted moving XOR task (XOR2), where feature contri-

Table 3.1. Comparison of feature selection on XOR1, XOR2, and MNIST. **Bold** indicates best performance and underline indicates second-best.

	XOR1	**XOR2**	**MNIST**
	Accuracy (%)	R^2 score	Accuracy (%)
LASSO	50.09 (1.23)	0.3109 (0.0175)	83.87 (0.05)
with context	50.05 (1.09)	0.3105 (0.0170)	83.88 (0.04)
Prediction model	70.74 (17.72)	0.3409 (0.0192)	98.11 (0.05)
with context	71.93 (11.88)	0.3382 (0.0167)	98.21 (0.07)
STG	73.98 (0.69)	0.3516 (0.0287)	98.55 (0.08)
with context	74.22 (1.25)	0.3487 (0.0269)	98.34 (0.07)
CEN	50.08 (0.67)	0.5186 (0.0126)	87.66 (0.24)
Contextual LASSO	50.60 (0.61)	0.7316 (0.0090)	97.17 (0.04)
AFS	78.95 (19.91)	0.7572 (0.0187)	94.83 (0.84)
INVASE	52.36 (1.70)	0.4627 (0.0194)	88.06 (2.30)
c-STG (Ours)	**100** (0)	0.8739 (0.0107)	98.66 (0.05)
Weighted c-STG (Ours)	**100** (0)	**0.9956** (0.0008)	**98.69** (0.06)

butions vary continuously with context:

$$\textbf{XOR2:} \quad y(\mathbf{x};\mathbf{z}) = \begin{cases} \mathrm{ReLU}(0.5x_1 + x_2), & z = 0, \\ \mathrm{ReLU}(x_1 + 0.5x_2), & z = 1, \\ \mathrm{ReLU}(0.5x_3 + x_4), & z = 2, \\ \mathrm{ReLU}(x_3 + 0.5x_4), & z = 3. \end{cases}$$

This setting requires not only identifying which features are relevant, but also capturing graded, context-dependent differences in feature importance. As shown in Fig. 3.3, standard c-STG correctly recovers the sparse structure of relevant features for each context, while weighted c-STG additionally captures the relative contribution of each feature through smoothly varying gate magnitudes. These qualitative differences translate into quantitative gains: Table 3.1 shows that weighted c-STG achieves the highest R^2 score on XOR2, outperforming both c-STG and all competing baselines. Moreover, Table 3.2 indicates that this improvement is achieved with the minimum possible number of selected features, highlighting the advantage of weighted c-STG

Table 3.2. Comparison of the number of selected features on XOR1, XOR2, and MNIST. **Bold** indicates the least number of selected features and underline indicates second-best.

	XOR1	**XOR2**	**MNIST**
	# features	# features	# features
LASSO	18.80 (0.87)	14.20 (2.04)	621.70 (4.77)
with context	21.50 (0.92)	14.60 (1.85)	627.5 (3.26)
STG	6.60 (1.50)	5.60 (1.02)	585.40 (3.77)
with context	4.20 (0.40)	4.20 (0.40)	**148.80** (2.64)
Contextual LASSO	4.60 ± 1.93	2.75 (0.93)	357.72 (8.88)
AFS	5.40 (4.27)	**2.00** (0.00)	290.00 (20.00)
INVASE	9.85(1.47)	13.04 (0.09)	376.49 (12.30)
c-STG (Ours)	**3.00** (0.00)	**2.00** (0.00)	247.25 (10.84)
Weighted c-STG (Ours)	**3.00** (0.00)	**2.00** (0.00)	217.02 (2.87)

when feature relevance varies continuously rather than discretely across contexts.

To illustrate that c-STG avoids coefficient shrinkage while maintaining context-specific sparsity, we introduce a linear synthetic task (XOR3) in which different subsets of features are relevant across contexts. Figure 3.4 compares feature importance estimates obtained using Contextual LASSO and c-STG. While Contextual LASSO assigns non-zero weights to many irrelevant features due to coefficient shrinkage, c-STG precisely isolates the true context-specific features.

Finally, we construct a task (XOR4) where the number of informative features itself varies with context. As shown in Fig. 3.5, c-STG correctly adapts both the identity and the number of selected features as a function of context, recovering context-dependent sparsity patterns without manual tuning. This behavior is not achievable by population-level baselines, which assume a fixed sparsity pattern across all contexts. Together, these controlled experiments demonstrate that c-STG and weighted c-STG recover context-dependent feature relevance accurately, achieve superior predictive performance, and do so with substantially fewer selected features than competing methods.

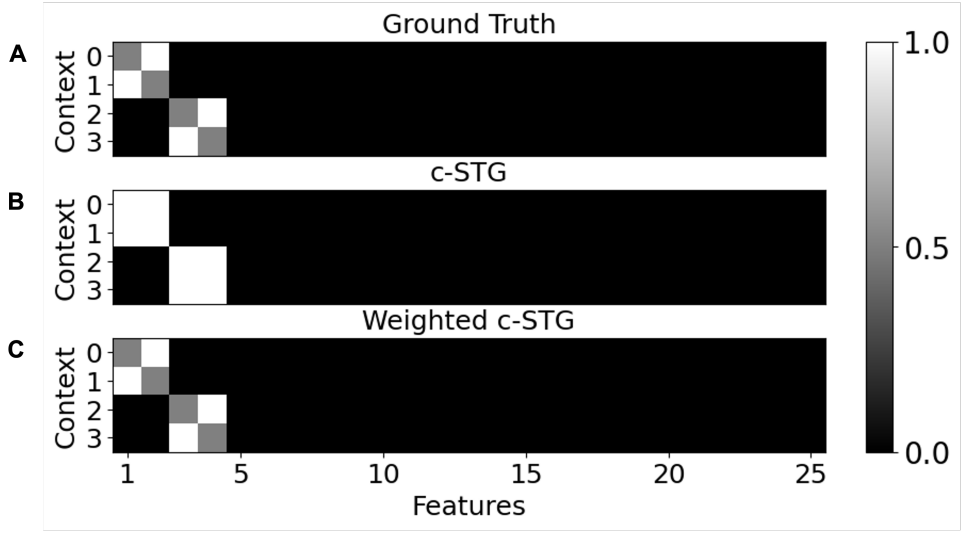


Figure 3.3. XOR2. (A) Ground-truth feature relevance as a function of context. (B) c-STG gates and (C) weighted c-STG gates, illustrating recovery of both sparsity and graded importance.

3.6.2 Image classification with contextual variability

We apply our proposed method for image classification using a subset of the MNIST dataset, including digit images of 4 and 9. To create a contextually diverse set of input images, we rotate each original image by intervals of 45 degrees, resulting in eight distinct images per original image. In this example, the goal is to identify which pixels most effectively differentiate between the digits per rotation angle. Applying c-STG or weighted c-STG where rotation serves as context results in a test accuracy of 98% (Table 3.1). In Fig. 3.6, we contrast STG with c-STG, illustrating the superiority of context-specific feature selection over population-level analysis. By comparing separate STG models for distinct categorical contexts against a unified c-STG model, we find that c-STG leads to higher accuracy with fewer samples (Fig. 3.8).

Sparsity-Driven Performance Analysis:

We evaluate the performance of c-STG as a function of the number of selected features in the model, i.e., sparsity. Our analysis uses the MNIST example, as described in Section 3.6.2. To perform this analysis, we varied the regularization coefficient, λ, in a way that controlled the sparsity level within the range of 10 to 50. It is worth noting that for STG with context, we only

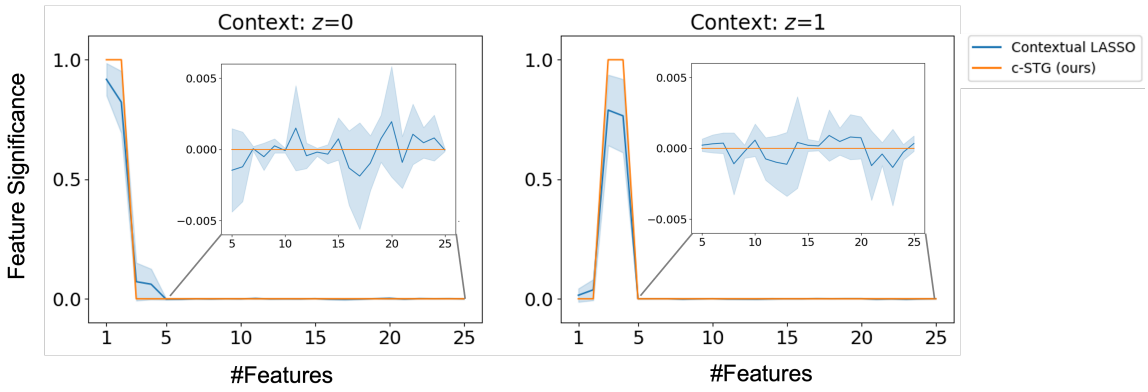


Figure 3.4. Comparison between Contextual LASSO and c-STG on a linear synthetic task. c-STG accurately isolates context-specific features, whereas Contextual LASSO assigns non-zero weights broadly due to shrinkage.

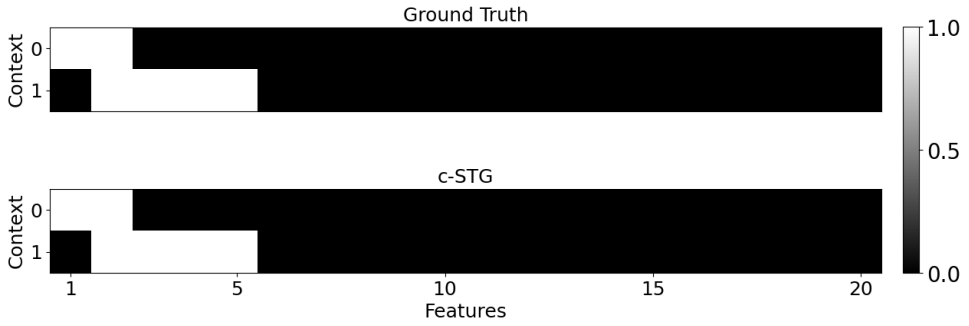


Figure 3.5. XOR4. Ground-truth feature relevance (top) and features selected by c-STG (bottom), demonstrating recovery of context-dependent sparsity patterns where both the identity and number of informative features vary with context.

counted the explanatory features corresponding to pixels and not the eight additional contextual features (one-hot encoding) representing the angle. Figure 3.7 illustrates the performance of three models, c-STG, STG, and STG with context as the sparsity level increases. This demonstrates that c-STG has superior performance compared to both STG with context and STG alone, indicating that c-STG, with its contextual information and sparse representation, outperforms the other models for varying numbers of selected features.

Comparison of Discrete Contextual Model Training:

When focusing on deriving insights from specific contextual features, training separate models for each specific context is a viable strategy. However, it is crucial to understand that

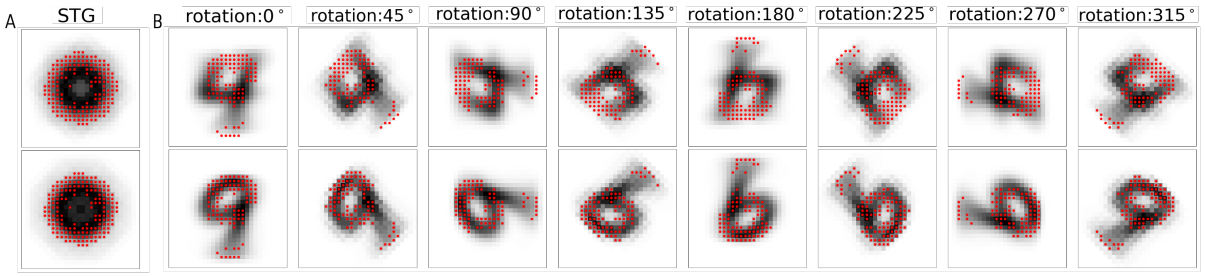


Figure 3.6. Comparison of pixel-level feature relevance for digit classification under rotational context. c-STG identifies rotation-specific informative pixels, whereas population-level STG selects global features.

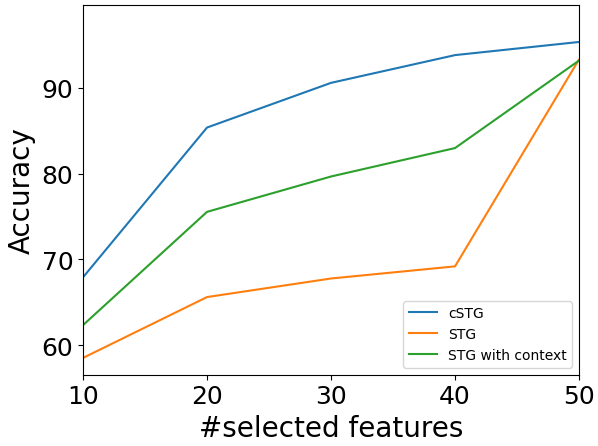


Figure 3.7. Comparing the performance variations of cSTG, STG, and STG with context with respect to different sparsity levels on the MNIST dataset.

this strategy is only applicable when the contextual variables are categorical and not continuous. Although straightforward, this method brings its own challenges, particularly when the dataset at hand is limited in size. In such cases, each model only gets to train on a subset of data corresponding to its designated context. We shed light on the consequences of this approach using the modified MNIST dataset, as elaborated in Section 3.6.2. Our exploration involved comparing two methodologies: 1) individually training eight models for each context using STG and 2) employing our c-STG to create a unified model for all contexts. To assess the efficiency of these approaches, we varied the number of training samples and captured the results in Fig. 3.8. Comparatively, our c-STG model offers a compelling advantage. Instead of segregating data by context, it utilizes the entirety of the dataset, benefiting from the diverse range of contexts.

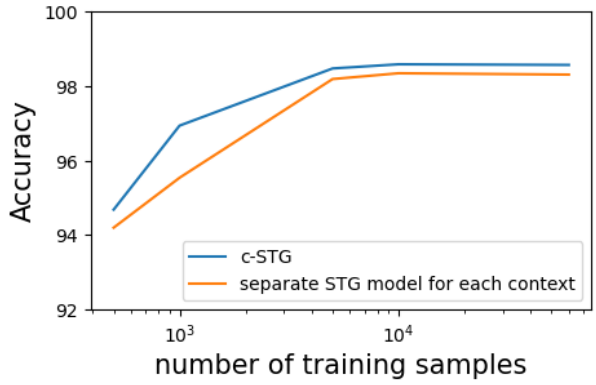


Figure 3.8. Comparison between training separate models for different values of categorical context variables and a single c-STG model on the modified MNIST dataset in Section 3.6.2, as the number of training samples varies.

A direct contrast in terms of model parameters reveals that while training separate models for each discrete context demands 876,688 parameters, the c-STG model operates with a lean set of 218,834 parameters. The reduced parameter count in c-STG does not compromise performance; in fact, it underscores c-STG's prowess in delivering enhanced outcomes with fewer parameters.

3.6.3 Time-dependent neuron relevance during reach outcome decoding

In studying the way the brain encodes behavior, perception, and memory, machine learning models are trained to predict measures of behavior from neuronal activity recordings. Feature selection can provide valuable insights by revealing individual (or subsets of) neurons encoding a particular behavior as a function of task-relevant timing, sensory input, or arousal state. In Levy et al. [2020], the authors study a motor task where mice are trained to perform a hand-reach task of a food pellet, given an auditory cue (tone). They recorded the activity of neurons in layers 2-3 of the primary motor cortex, where each trial was labeled as successful if the animal managed to grab and consume the food pellet. Cellular networks in the motor cortex are expected to communicate error signals while acquiring a new motor task to achieve improvement across attempts. To test this hypothesis, Levy et al. [2020] trained a separate SVM binary model to classify trials as success or failure per neuron based on the activity during a short

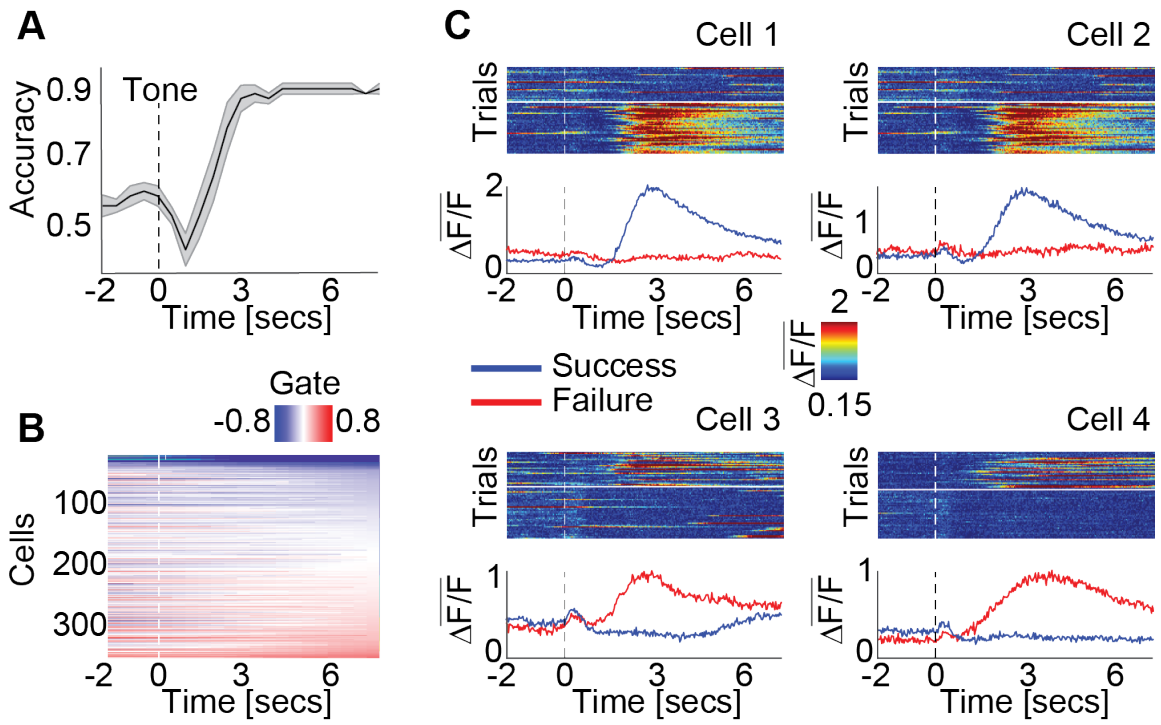


Figure 3.9. Neuroscience: Context is time. (A) Prediction accuracy as a function of time from cue onset. (B) Weighted c-STG gate values across neurons over time. (C) Example neurons exhibiting outcome-selective activity, showing prolonged responses on either success or failure trials.

time window. This resulted in training 7866 models (342 neurons $\times$ 23 sliding time windows). The analysis showed that 1) outcome can be consistently decoded from neuronal activity starting 2 seconds after the tone till the end of the trial; 2) 12% of neurons exhibited prolonged activity starting 2 seconds after the tone, on either success or failure trials, thus "reporting" trial outcome.

Using a single c-STG model, we reveal the same biological findings. We set the prediction model to be a binary classifier (success or failure); the explanatory features are the neuronal activity of all neurons in a given time window, and the contextual variable is *time*. Fig. 3.9(A) presents the test accuracy of weighted c-STG as a function of time, indicating that the trial outcome was successfully classified starting from 2 seconds after the tone. The learned weights are presented in (B), where 12% of cells have absolute weight values that are larger than 0.25. Four example neurons with high absolute weights (C) exhibit prolonged activity only during

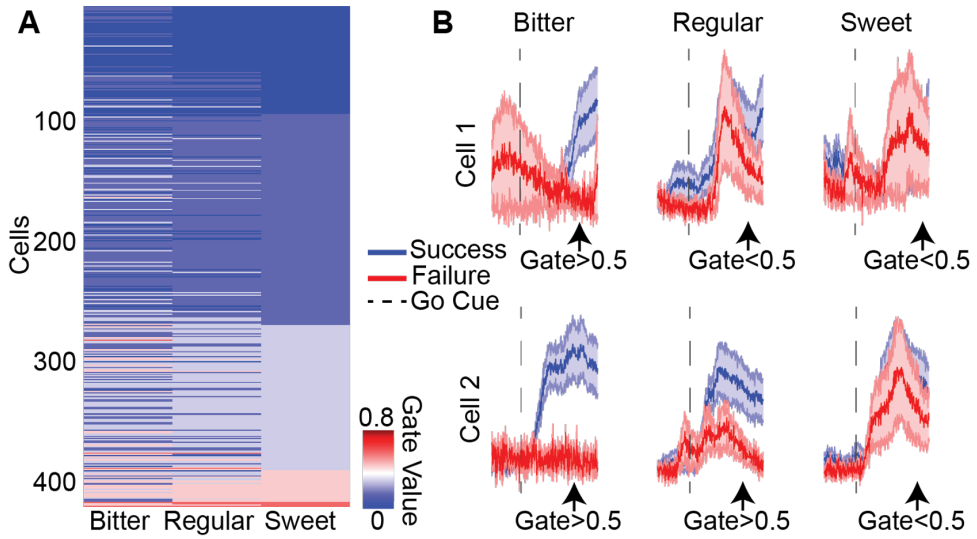


Figure 3.10. Neuroscience: Context is flavor. (A) c-STG gate values across neurons for each flavor condition. (B) Example neurons selected by c-STG, illustrating flavor-specific differences in outcome-related activity for successful and failed trials.

either success or failure. Overall, weighted c-STG captured the complex dynamics of how outcome is encoded within the ensemble neuronal activity; it was able to detect individual neurons reporting outcome as a function of time using a single model, a biological finding which Levy et al. [2020] used thousands of SVM models to reveal.

3.6.4 Flavor-dependent neuron relevance in outcome decoding

We apply c-STG to another hand-reach study where a mouse was given flavored food pellets: bitter, sweet, or regular (unflavored). We hypothesize that outcome encoding depends on flavor. We train c-STG to classify the outcome based on the activity in the last 4 seconds of each trial, where the contextual variable is *flavor*. Our results in Fig. 3.10 reveal that the cellular network encodes the outcome differently across flavors. We quantify how similar the encoding is by correlating the gate values across flavors and find they are most different for sweet and bitter (c=0.4), and they are more similar for regular and sweet (c=0.7) than for regular and bitter (c=0.6). Overall, c-STG is able to capture the complexity of outcome encoding by the neuronal network across different contexts.

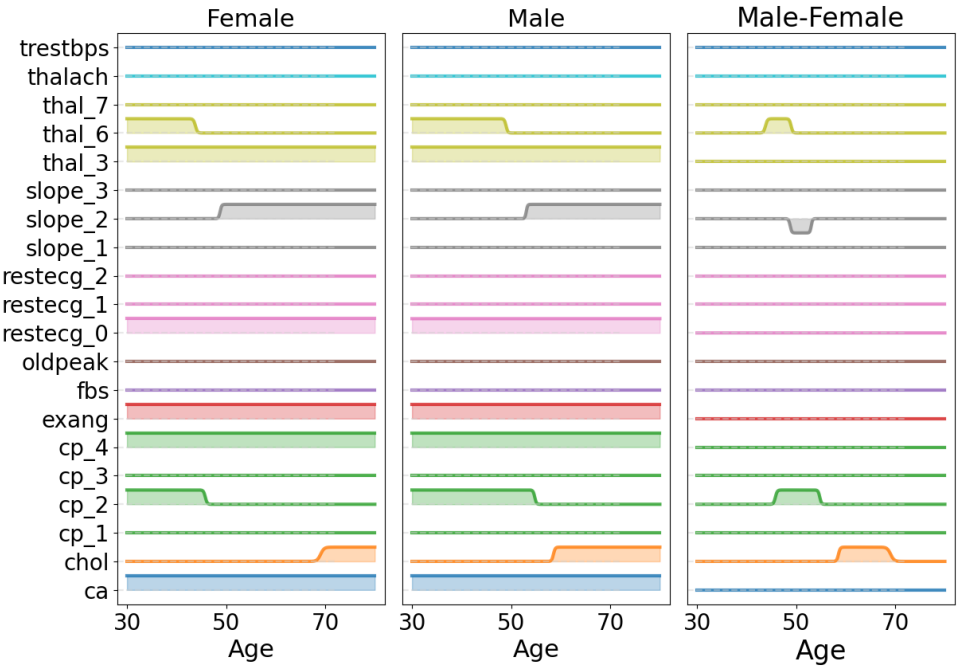


Figure 3.11. Heart disease. Feature selection gates $\mu(z)$ for each input feature as a function of context—age and gender (left - females and center - males)—produced by weighted c-STG. The difference in the c-STG values between males and females indicates gender-specific informative features as a function of age.

3.6.5 Age and gender dependent risk factors for heart disease

We now focus on medical data, specifically, the heart disease dataset from UCI ML repository Janosi et al. [1988]. Given features including chest pain type, resting blood pressure, serum cholesterol levels, and more, our goal is to understand how age and gender influence these biometrics in relation to the risk of heart disease. Age and gender are set as a two-dimensional contextual input to our hypernetwork in this binary classification problem. Predictive performance is summarized in Table 3.3, where weighted c-STG achieves the best accuracy while selecting the fewest features among competing sparse baselines.

Weighted c-STG analysis (Fig. 3.11) reveals age and gender-specific feature significance for heart disease. For example, cholesterol ('chol') monitoring becomes crucial with increased age as a risk factor for cardiovascular disease. In males, cholesterol's relevance arises around age 50, likely due to the absence of hormone-induced cholesterol regulation. In females, a significant

Table 3.3. Comparison of prediction performance and number of selected features on Heart disease and Housing datasets. **Bold** indicates best performance (or least number of features), and underline indicates second-best.

	Heart disease		**Housing**	
	Accuracy (%)	# features	R^2 score	# features
LASSO	83.50 (3.69)	13.6 (1.11)	0.2161 (0.0014)	8.00 (0.00)
with context	83.67 (3.48)	16.90 (0.70)	0.2272 (0.0016)	10.00 (0.00)
Prediction model	86.53 (1.51)	–	0.2022 (0.0361)	–
with context	83.92 (5.52)	–	0.2132 (0.0381)	–
STG	86.53 (2.11)	16.60 (2.65)	0.2139 (0.0023)	8.00 (0.00)
with context	87.88 (1.95)	18.00 (4.15)	0.2234 (0.0022)	8.00 (0.00)
CEN	84.17 (5.69)	11.79 (3.66)	0.2561 (0.0002)	5.43 (0.29)
Contextual LASSO	83.19 (5.32)	11.79 (3.66)	0.4616 (0.0033)	5.43 (0.29)
AFS	86.44 (1.07)	8.00 (5.48)	0.3963 (0.0092)	**2.70** (0.64)
INVASE	87.22 (2.20)	9.12 (1.85)	0.3335 (0.0203)	4.80 (0.16)
c-STG (Ours)	87.55 (3.41)	9.69 (0.81)	0.3976 (0.0082)	**2.70** (0.42)
Weighted c-STG (Ours)	**89.23** (1.72)	**6.50** (0.50)	**0.5308** (0.0052)	4.36 (0.30)

cholesterol risk arises later, past age 70, when they are post-menopause, and estrogen's protective effect has faded. Similarly, a flat ST segment slope ('slope_2') indicates heart disease risk in females starting at age 45, intensifying post-menopause. For males, 'slope_2' becomes significant after 55, showcasing distinct cardiovascular risk patterns between genders.

3.6.6 Location-dependent socio-economic feature relevance

In the domain of housing price prediction, understanding how various features, such as the renovation condition or floor height, influence its cost based on the house's location in the city requires context-specific feature selection. The Housing dataset Lianjia [2017] includes a set of features such as age, elevator, floor height, renovation condition, and the latitude and longitude of the house. We used our proposed approach to predict the price of the house using all the features while considering the location (latitude and longitude) as two-dimensional contextual data. For this dataset, our weighted c-STG outperforms all other models in R^2 score (Table 3.3). Visualizing weighted c-STG feature significance in Fig. 3.12 offers key insights into the Housing dataset. First, feature importance shows a localized pattern, with neighboring locations displaying similar feature significance. Secondly, subway presence in the city center positively impacts house

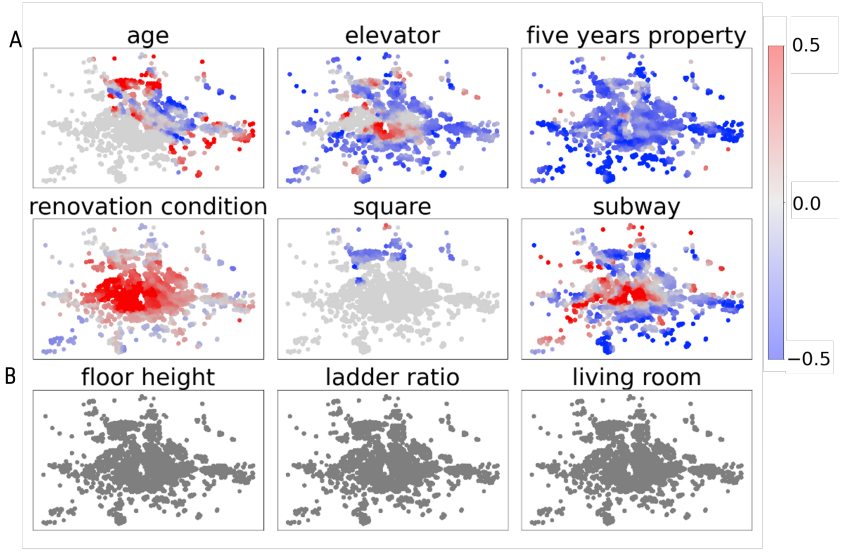


Figure 3.12. Housing. Geographic significance of nine housing features according to weighted c-STG ($w(z) \times \widetilde{s}(z)$). The red-to-blue gradient denotes positive to negative impact, respectively, with gray indicating negligible contribution. (A) Spatial distribution of selected features. (B) Features not selected by weighted c-STG.

pricing, possibly due to enhanced accessibility and convenience. In contrast, subway presence negatively affects prices in the outskirts, potentially due to increased noise and disruption. Additionally, renovated properties in the city center positively influence pricing, likely because renovations in high-demand urban areas add significant value to properties. These findings provide valuable insights for real estate companies and policymakers. By understanding which features are more important in which locations, they can make informed decisions regarding housing prices, regulations, and development.

3.6.7 Performance under controlled sparsity

We next examine feature selection behavior across methods, focusing on how predictive performance varies when the number of selected features is explicitly controlled. While Tables 3.1 and 3.3 report performance under each method's preferred sparsity level, this analysis isolates the effect of feature selection by fixing the number of selected features across models.

Tables 3.4 and 3.5 compare predictive performance when all methods are constrained to select the same number of features for each dataset. The fixed sparsity levels (reported in the

Table 3.4. Comparison of feature selection techniques with a fixed number of significant features across all the models on XOR and MNIST datasets. The number of significant features selected for each dataset is reported in the second row. **Bold** indicates best performance and underline indicates second-best.

	XOR1	**XOR2**	**MNIST**
	3	2	218
	Accuracy (%)	R^2 score	Accuracy (%)
LASSO	50.07 (1.43)	0.1760 (0.0156)	73.70 (0.48)
with context	50.02 (1.09)	0.1761 (0.0156)	73.79 (0.48)
STG	65.73 (1.46)	0.1838 (0.0248)	91.47 (0.67)
with context	65.77 (0.87)	0.1873 (0.0320)	98.34 (0.07)
Contextual LASSO	50.53 (0.60)	0.7311 (0.0090)	97.08 (0.06)
AFS	70.83 (19.06)	0.7572 (0.0187)	93.91 (0.99)
INVASE	50.07 (0.70)	0.3547 (0.0118)	65.27 (4.69)
c-STG (Ours)	**100** (0)	0.8739 (0.0107)	98.65 (0.04)
Weighted c-STG (Ours)	**100** (0)	**0.9956** (0.0008)	**98.69** (0.06)

Table 3.5. Comparison of feature selection techniques with a fixed number of significant features across all the models on Heart disease and Housing datasets. The number of significant features selected for each dataset is reported in the second row. **Bold** indicates best performance and underline indicates second-best.

	Heart disease	**Housing**
	7	5
	Accuracy (%)	R^2 score
LASSO	73.70 (0.48)	0.1979 (0.0016)
with context	82.00 (3.48)	0.2251 (0.0015)
STG	84.18 (2.92)	0.2136 (0.0026)
with context	83.84 (1.31)	0.2231 (0.0022)
Contextual LASSO	82.85 (4.83)	0.4535 (0.0034)
AFS	83.39 (1.98)	0.3482 (0.0224)
INVASE	84.18 (1.72)	0.2502 (0.0060)
c-STG (Ours)	86.89 (3.18)	0.3970 (0.0074)
Weighted c-STG (Ours)	**89.23** (1.72)	**0.5308** (0.0052)

second row of each table) are chosen to match the number of features selected by c-STG in the unconstrained setting. This allows for a fair comparison that decouples predictive accuracy from differences in sparsity.

Across all datasets, c-STG and weighted c-STG achieve the best or second-best performance under fixed sparsity constraints. On the synthetic XOR tasks and MNIST (Table 3.4), c-STG attains perfect accuracy on XOR1 and near-optimal performance on XOR2 and MNIST, outperforming population-level methods such as LASSO and STG, as well as contextual baselines including Contextual LASSO and AFS. Notably, several competing methods exhibit substantial performance degradation when restricted to the same number of features, indicating reduced robustness to sparsity constraints.

A similar pattern is observed on real-world datasets (Table 3.5). For Heart disease prediction, weighted c-STG achieves the highest accuracy under fixed sparsity, while for Housing price prediction it substantially outperforms all baselines in terms of R^2. These results demonstrate that the performance gains of c-STG are not solely due to selecting more features, but instead arise from its ability to identify the most informative features in a context-dependent manner.

We exclude the prediction-only model (with or without context) from this sparsity analysis, as no feature selection or sparsity constraint is imposed on its explanatory variables. Overall, these results confirm that c-STG maintains strong predictive performance even under strict sparsity constraints, highlighting its effectiveness as an interpretable and context-aware feature selection method.

3.6.8 Implementation details

In selecting model architecture, we aim to maintain simplicity, interpretability, and linearity whenever feasible. We retained the same hypernetwork and prediction model architecture across various methods when possible. For the hypernetwork, we conducted experiments with one and two hidden layers, while for the prediction networks, we explored options ranging from zero (linear) to two hidden layers. The final model architectures for each of the datasets are mentioned

below in their respective paragraphs. To determine the best hyperparameters, namely the learning rate (η) and regularization coefficient (λ), we performed a grid search over the following values: $\eta \in \{1e^{-1}, 5e^{-2}, 1e^{-2}, 5e^{-3}, 1e^{-3}, 5e^{-4}, 1e^{-4}\}$ and $\lambda \in \{1, 5e^{-1}, 1e^{-1}, 5e^{-2}, 1e^{-2}, 5e^{-3}, 1e^{-3}\}$. The same set of values was used for the grid search across all the datasets. The selection of model parameters/hyperparameters was based on preventing issues like underfitting and overfitting and ensuring optimal 5-fold cross-validated performance.

XOR1:

The hypernetwork $\widetilde{h}_\phi$ has two fully connected layers, with 100 and 10 neurons, respectively. We employed ReLU as the activation function after the first layer and Sigmoid activation after the last layer. For the prediction model, we used two fully connected layers, with 10 and 10 neurons, respectively, and their corresponding nonlinearities, ReLU and sigmoid.

XOR2:

Both the hypernetwork $\widetilde{h}_\phi$ and the prediction model have no hidden layers with inputs directly projecting to outputs. However, the hypernetwork has a Sigmoid activation, and the prediction model has a ReLU activation function in their output layers.

XOR3:

For the XOR3 dataset, the architecture of hypernetwork and prediction model is same as that of XOR2 except that the prediction model has no activation function.

XOR4:

For the XOR4 dataset, the architecture of hypernetwork and prediction model is same as that of XOR3.

MNIST:

For the MNIST dataset, the configuration of the hypernetwork $\widetilde{h}_\phi$ differs, possessing two layers with 64 and 128 neurons. ReLU and Sigmoid activation functions are used for these two

layers, respectively. The prediction model consists of layers with 128 and 64 neurons, coupled with the ReLU followed by Sigmoid activations.

Housing:

We employed a hypernetwork architecture similar to the one used in the MNIST example. However, we used a linear model to predict house prices in this case. We divided the data into 10 train, validation, and test splits and conducted a grid search on η and λ for each split. We then selected the hyperparameters with the best validation performance.

Heart disease

The full list of features includes: chest pain type (cp), resting blood pressure (trestbps), serum cholesterol levels (chol), fasting blood sugar (fbs), resting electrocardiographic results (restecg), maximum heart rate achieved (thalach), exercise-induced angina (exang), ST depression induced by exercise (oldpeak), the slope of the peak exercise ST segment (slope), the number of major vessels colored by fluoroscopy (ca), and thalassemia (thal). Among these features, cp, restecg, and slope are categorical and, hence, are converted to one hot encoding vector for feeding them into the prediction model. The hypernetwork and the prediction model have one hidden layer with 1000 neurons and 10 neurons, respectively, with sigmoid activation on their last layer. We conducted a grid search on η and λ from the range of values mentioned in the MNIST example.

Neuroscience:

For the first data set (context is time), a weighted c-stg model was trained using 5-fold cross-validation with separate sets of trials for training, parameter tuning, and testing. We conducted a grid search on different learning rates and regularization parameters ranging from 0.0005 to 0.1 and for the number of hidden units of the hypernetwork ranging from 10 to 1000 units. Overall, for the hypernetwork, we used a single hidden layer of 10 units, a learning rate of 0.0005, and a regularization weight of 0.05, where the prediction network consisted of six hidden layers of 500, 300, 100, 50, and 2 units. For the second data set (flavor is context), a c-stg

model was trained using 5-fold cross-validation with the same mechanism for parameter tuning. Overall, for the hypernetwork, we used a single layer with 1000 units, a learning rate of 0.05, and a regularization weight of 0.001, where the prediction network consisted of six hidden layers of 400, 300, 100, 50, and 2 units.

Acknowledgements

Chapter 3, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Lindenbaum, Ofir; Lifshitz, Shira; Lavzin, Maria; Schiller, Jackie; Mishne, Gal; Benisty, Hadas (2024). "Contextual Feature Selection with Conditional Stochastic Gates." In *Proceedings of the 41st International Conference on Machine Learning (ICML)*. The dissertation author is the primary author of this paper.

Chapter 4

Coupled Transformer Autoencoder for Disentangling Multi-Region Neural Latent Dynamics

4.1 Motivation and related work

High-density electrophysiology (e.g., Neuropixels) and volumetric calcium imaging now enable simultaneous recordings from multiple brain regions at large scale, shifting population analysis from single-area snapshots to the level of *distributed circuits* [Jun et al., 2017, Machado et al., 2022]. These datasets reveal a coexistence of globally coordinated activity and region-specific specialization: individual neurons both receive input from and project to distant areas, and behaviorally relevant codes are often hypothesized to be broadcast across widespread circuits rather than confined to a single locus [Machado et al., 2022]. This distributed perspective raises a core mechanistic question: which computations are local, which are shared, and how are global brain states coordinated across interacting populations? Critically, answering this requires disentangling *shared* components that mediate inter-regional coordination from *private* signals unique to each region, a distinction that also guides the design and interpretation of causal experiments such as targeted perturbations and pathway-specific manipulations.

Latent variable models provide a principled framework for studying such population-level computation. Even within a single area, neural activity reflects a mixture of structured responses

and substantial trial-to-trial variability [Cunningham and Yu, 2014b], motivating the neural latent dynamics view in which high-dimensional population activity is governed by low-dimensional trajectories evolving over time [Vyas et al., 2020, Churchland and Shenoy, 2024]. Classical linear methods such as PCA [Pearson, 1901] and factor analysis (FA) [Harman, 1976] uncover low-dimensional structure and remain widely used in large-scale recordings [Cunningham and Yu, 2014b], but they do not explicitly model temporal continuity. To address this, Gaussian-Process Factor Analysis (GPFA) imposes smoothness through a GP prior [Yu et al., 2009], while nonlinear dynamical latent models such as LFADS [Pandarinath et al., 2018] and attention-based sequence models such as the Neural Data Transformer (NDT) [Ye and Pandarinath, 2021] relax linearity and improve expressivity for non-stationary dynamics. A broad line of work further targets identifiability and regime changes via switching state-space models [Linderman et al., 2016, Glaser et al., 2020], GP-SLDS variants [Hu et al., 2024], locally linear manifold structure [Abbaspourazad et al., 2024], and variational neural state-space models [Hernandez et al., 2018, Gu et al., 2022].

However, applying single-region latent models naïvely to multi-region recordings—e.g., by concatenating activity across areas—can obscure meaningful structure. Inter-area delays can warp the learned latent trajectories, differences in correlation structure can cause shared factors to absorb region-specific variance, and larger or more active populations can dominate the learned representation. These issues motivate models that explicitly represent inter-regional coordination. Complementary to predictive analyses of inter-areal coupling [Zandvakili and Kohn, 2015, Semedo et al., 2019, Perich et al., 2018], joint latent-variable approaches aim to decompose population activity into shared and private components. A prominent class relies on GP-based multi-group extensions, including Delayed Latents Across Groups (DLAG) [Gokcen et al., 2022], its multi-population generalization mDLAG [Gokcen et al., 2023], multi-view GPFA extensions [Gokcen et al., 2024], and Multi-Region Markovian Gaussian Process models [Li et al., 2024]. While these methods enforce temporal smoothness, they inherit GP stationarity assumptions and typically use linear read-outs, limiting their ability to capture non-

stationary or long-range nonlinear circuit dynamics. Classical multiset CCA [Hotelling, 1936] and tensor decomposition methods [Cichocki et al., 2015] also provide linear tools for multi-area recordings, while mechanistic RNN-based approaches such as CURBD infer directed currents between many regions but do not explicitly enforce a shared–private decomposition [Perich et al., 2020]. Together, these strands support the empirical view that inter-regional communication is mediated by a selective low-dimensional subset of population activity, motivating the concept of a *communication subspace* distinct from private dynamics [Semedo et al., 2019].

In parallel, multi-view representation learning in machine learning has developed coupled and shared–private autoencoder objectives that move beyond classical CCA. Deep CCA (DCCA) maximizes correlation between learned representations [Andrew et al., 2013], while Deep CCA Autoencoders (DCCAE) add reconstruction at the risk of mixing private and shared information [Wang et al., 2016]. Related families include deep coupled autoencoders [Wang et al., 2014], domain-adaptation coupled autoencoders [Wang and Breckon, 2023], correlation-based autoencoders for cross-modal retrieval [Feng et al., 2014], split-brain and shared–private disentangling objectives [Zhang et al., 2017, Bousmalis et al., 2016], and cycle-consistent multi-view autoencoders [Wu et al., 2018]. Recent disentangling frameworks such as DMVAE and SPLICE further partition shared and private factors [Lee and Pavlovic, 2021, Koukuntla et al., 2024], and multimodal VAEs learn shared/private structure across modalities via product- or mixture-of-experts formulations [Wu and Goodman, 2018, Shi et al., 2019, Sutter et al., 2021]. Yet, these multi-view methods typically treat samples as i.i.d. (or short windows), discarding the long-horizon temporal dependencies central to neural population dynamics, and several incur unfavorable scaling with the number of views or assume a single global shared component that precludes subset-specific shared latents [Lee and Pavlovic, 2021, Koukuntla et al., 2024].

Taken together, existing approaches fall short of jointly satisfying three requirements for distributed neural circuit analysis: *(i)* modeling non-stationary, nonlinear temporal dynamics; *(ii)* respecting temporal continuity in population activity; and *(iii)* scaling beyond two regions without a parameter explosion [Gokcen et al., 2022, 2023, 2024, Li et al., 2024, Koukuntla et al., 2024,

Lee and Pavlovic, 2021]. These gaps motivate the *Coupled Transformer Autoencoder* (CTAE), which uses transformer-based encoders and decoders [Vaswani et al., 2017] to capture long-range nonlinear dynamics while explicitly partitioning each region's representation into orthogonal shared and private subspaces. By combining reconstruction-based training with lightweight alignment and disentanglement regularizers, CTAE aims to recover behavior-agnostic latent embeddings that support diverse downstream decoding tasks using simple linear read-outs, while extending naturally to multi-region settings without exponential growth in parameters.

4.2 Problem formulation

Let $\boldsymbol{X}^{(1)} \in \mathbb{R}^{N_1 \times T}, \boldsymbol{X}^{(2)} \in \mathbb{R}^{N_2 \times T}$ denote simultaneous neural population recordings from two brain regions, each acquired over T time steps and, N_1 and N_2 channels respectively (generalization to more than 2 regions is in section 4.4).

We assume that the observed neural activity at each time step t is a nonlinear transformation of latent variables (Fig.4.2), which include: (i) shared latent dynamics within a communication subspace [Semedo et al., 2019] spanned by neural trajectories that are correlated across regions, and (ii) private latent dynamics that capture region-specific processes lying in subspaces orthogonal to the shared component. To effectively capture the temporal structure of neural activity, we model these latent variables as functions of the entire observed neural activity.

$$\mathbf{X}_t^{(1)} = f\big(\boldsymbol{S}_{1:t}, \boldsymbol{P}_{1:t}^{(1)}\big), \qquad \mathbf{X}_t^{(2)} = g\big(\boldsymbol{S}_{1:t}, \boldsymbol{P}_{1:t}^{(2)}\big) \qquad \text{for } t \in \{1, \ldots, T\} \tag{4.1}$$

where $1:t$ denotes the sequence of indices between 1 and t, $\boldsymbol{S}_t \in \mathbb{R}^{d_s}$ represents the *shared* dynamics between the two regions at time t, and $\boldsymbol{P}_t^{(1)} \in \mathbb{R}^{d_1}$ and $\boldsymbol{P}_t^{(2)} \in \mathbb{R}^{d_2}$ capture dynamics specific to regions 1 and 2, respectively, and d_s, d_1, and d_2 denote the dimensionalities of the shared and private representations.

Our goal is to recover the shared and private latent dynamics $\boldsymbol{S}$, $\boldsymbol{P}^{(1)}$, and $\boldsymbol{P}^{(2)}$ given the observed neural activity $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$.

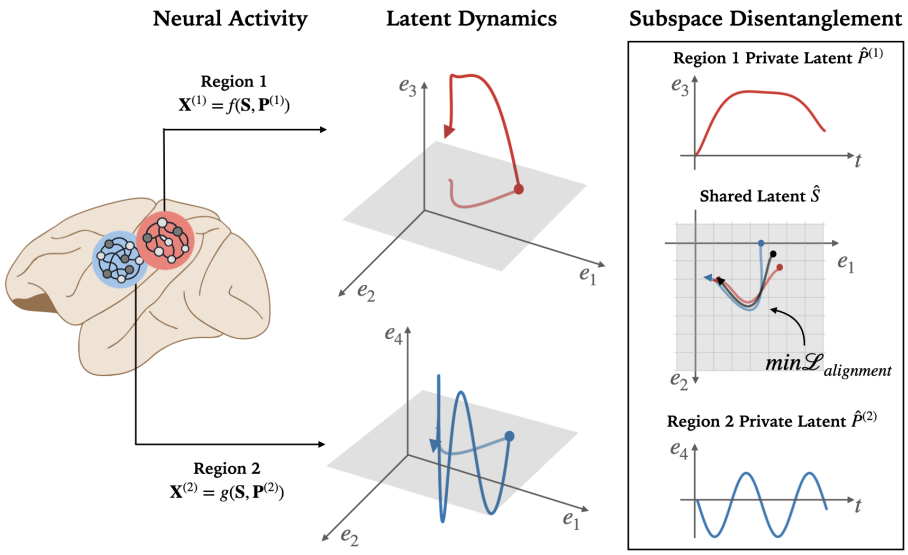


Figure 4.1. Observed neural activity across time from two brain regions, denoted as $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$, is modeled as a nonlinear function of underlying latent dynamics specific to each region. In the illustration, $[e_1, e_2, e_3]$ span the latent subspace for region 1 and $[e_1, e_2, e_4]$ for region 2. Inter-regional communication is mediated by shared latent trajectories S within the common subspace $[e_1, e_2]$, which drive correlated population activity. Following the output-null/potent hypothesis, we assume shared and private dimensions are orthogonal, allowing clear recovery of shared dynamics from region-specific processes.

Notations. Bold upper-case letters denote matrices (e.g., $\mathbf{X}$), bold lower-case denote latent vectors (e.g., $\boldsymbol{w}$), and $\widehat{\cdot}$ indicates reconstructions. Let $\mathbf{1}_n$ denote a vector of all-ones of length n and let $\mathbf{0}_n$ denote a vector of all zeros of length n. $\|(\cdot)\|_F$ denotes the Frobenius norm. $\odot$ denotes element-wise multiplication. Expectation is over batches of trials unless stated otherwise.

4.3 CTAE for two-region neural recordings

Our model is comprised of a coupled autoencoder built on transformer models, that disentangles shared and private representations. We have designed loss functions to recover latents that maximally represent the neural activity, where the shared representations are aligned across all regions, and each disentangled sub-space is orthogonal to the others. For clarity, we describe the two-region case in the main text; the extension to an arbitrary number of regions is provided in Appendix 4.4.

4.3.1 Model Architecture

We design separate causal Transformer-based encoder–decoder pairs, denoted by $(E_\theta^{(1)}, D_\phi^{(1)})$ and $(E_\theta^{(2)}, D_\phi^{(2)})$, for each of the two brain regions. Each encoder is a Transformer stack with *self-attention* layers that capture long-range, nonlinear temporal dependencies within a region. The recorded spike trains are Gaussian-smoothed to obtain continuous firing-rate estimates which are the inputs to the encoder. Each decoder employs standard Transformer *cross-attention* to reconstruct the original firing rates from its region's latents. Processing the full multichannel time series from each session, the encoders produce latent representations partitioned into two distinct components: shared latent sequences capturing dynamics common to both regions and private latent sequences specific to each region:

$$\mathbf{Z}^{(1)} = E_\theta^{(1)}\big(\mathbf{X}^{(1)}\big), \qquad \mathbf{Z}^{(2)} = E_\theta^{(2)}\big(\mathbf{X}^{(2)}\big), \qquad \mathbf{Z}^{(1)}, \mathbf{Z}^{(2)} \in \mathbb{R}^{D\times T} \tag{4.2}$$

Here D is the total number of latent dimensions. To indicate which of the latent dimensions correspond to shared and private representations, we introduce weight vectors $\mathbf{w}_1$ and $\mathbf{w}_2 \in \{0,1\}^D$.

Region–specific weight masks. Let the latent dimension be partitioned as $D = d_s + d_1 + d_2$. Define three contiguous index sets $\mathscr{I}_s = \{1,\ldots,d_s\}$, $\mathscr{I}_1 = \{d_s{+}1,\ldots,d_s{+}d_1\}$, and $\mathscr{I}_2 = \{d_s{+}d_1{+}1,\ldots,D\}$. We then construct binary weight vectors that indicate which subset of the latent dimensions correspond to the shared or to region-private. Define $\mathbf{w}_1, \mathbf{w}_2 \in \{0,1\}^D$ as

$$\mathbf{w}_1 = \big[\underbrace{\mathbf{1}_{d_s}}_{\mathscr{I}_s}, \underbrace{\mathbf{1}_{d_1}}_{\mathscr{I}_1}, \underbrace{\mathbf{0}_{d_2}}_{\mathscr{I}_2}\big]^\top, \qquad \mathbf{w}_2 = \big[\underbrace{\mathbf{1}_{d_s}}_{\mathscr{I}_s}, \underbrace{\mathbf{0}_{d_1}}_{\mathscr{I}_1}, \underbrace{\mathbf{1}_{d_2}}_{\mathscr{I}_2}\big]^\top. \tag{4.3}$$

so that $\mathbf{w}_1$ activates the shared and region-private of region 1 dimensions, whereas $\mathbf{w}_2$ activates the shared and region-private dimensions of region 2. Throughout the paper we treat $\mathbf{w}_r$ as fixed; the latent space dimensions (d_s, d_1, d_2) are treated as hyper-parameters and tuned on a validation set. Importantly, these masks do not hard-code the actual interaction structure: they

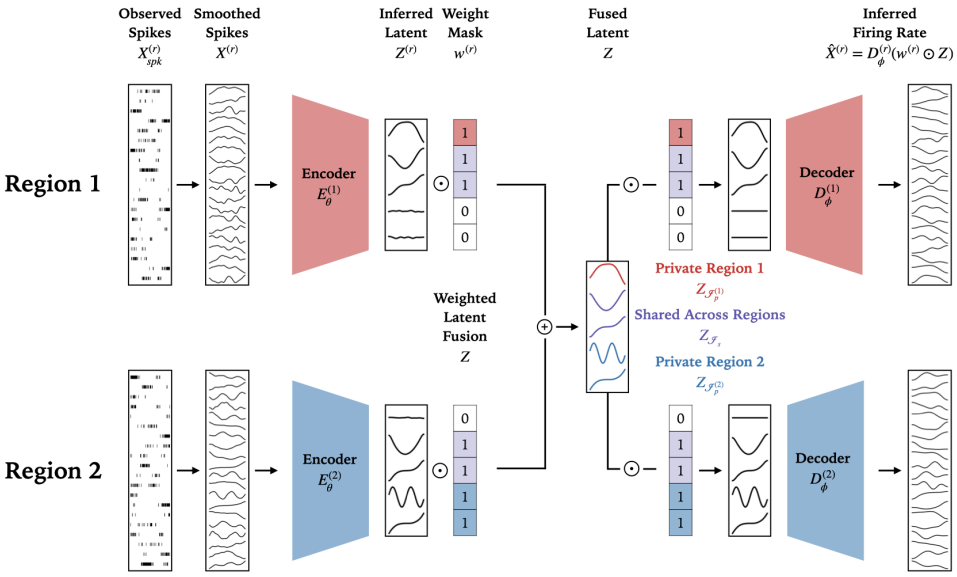


Figure 4.2. CTAE architecture. CTAE is composed of a coupled autoencoder, where the encoders and decoders are causal transformers designed to reconstruct neural activity for each region r. The inputs to the network are estimated spike rates from each region. A weight mask per region $w^{(r)}$ is used to disentangle the shared representation (violet) from the region-specific latents (red and blue) in the encoder outputs $Z^{(r)}$. The latents are recovered via end-to-end training.

only specify an *upper bound* on how many shared or private latents the model may allocate. During training, dimensions unsupported by the data naturally collapse to negligible variance. While fixed masks provide clarity and control in the two-region setting, they could also be initialized from anatomical priors or made learnable so that the model itself infers which latents are shared versus private. We leave these scalable extensions to future work.

Weighted latent fusion. Broadcasting the masks across time, we aggregate the two latent trajectories dimension-wise via a masked average for each dimension d in the latent:

$$\mathbf{Z}_t[d] = \frac{\mathbf{w}_1[d]\mathbf{Z}_t^{(1)}[d] + \mathbf{w}_2[d]\mathbf{Z}_t^{(2)}[d]}{\mathbf{w}_1[d] + \mathbf{w}_2[d]}. \tag{4.4}$$

Equation (4.4) leaves the private blocks unchanged (they receive weight 1 from only their own region)

$$\widehat{\boldsymbol{P}}^{(1)} = \mathbf{Z}_{\mathscr{I}_1} = \mathbf{Z}^{(1)}_{\mathscr{I}_1}, \quad \widehat{\boldsymbol{P}}^{(2)} = \mathbf{Z}_{\mathscr{I}_2} = \mathbf{Z}^{(2)}_{\mathscr{I}_2}. \tag{4.5}$$

while averaging the shared block across regions:

$$\widehat{\boldsymbol{S}} = \mathbf{Z}_{\mathscr{I}_s} = \tfrac{1}{2}\big(\mathbf{Z}^{(1)}_{\mathscr{I}_s} + \mathbf{Z}^{(2)}_{\mathscr{I}_s}\big). \tag{4.6}$$

This design will implicitly force alignment of the shared latents of both regions, via minimizing the reconstruction losses we define in the next section.

Region-specific decoding. Each decoder receives only the latent dimensions relevant to its own region:

$$\widehat{\mathbf{X}}^{(r)} = D^{(r)}_{\phi}\big((\mathbf{w}_r \mathbf{1}_T^{\top}) \odot \mathbf{Z}\big), \tag{4.7}$$

where $(\mathbf{w}_r \mathbf{1}_T^{\top}) \in \mathbb{R}^{D \times T}$ is an outer product such that each weight is duplicated for each dimension along all time points. Thus, the element-wise product zeroes out unrelated latents, forcing decoder r to rely exclusively on the subset of dynamics meaningful for its region.

This architecture (i) aligns both regions in a common latent space of dimension D, (ii) preserves region-specific structure via fixed relevance masks, and (iii) enforces cross-region consistency through the fusion rule in Eq. (4.4). The loss function balancing reconstruction accuracy and alignment objectives is detailed in Section 4.3.2. We illustrate our architecture in Fig. 4.2.

4.3.2 Training Objective

We use four loss functions to recover the shared and private latent representations with CTAE:

Reconstruction loss. The first loss is a reconstruction loss, so that each transformer autoencoder faithfully reproduces its own region's activity:

$$\mathscr{L}_{\text{rec}} = \sum_{r=1}^{2} \big\| \widehat{\mathbf{X}}^{(r)} - \mathbf{X}^{(r)} \big\|_F^2. \tag{4.8}$$

Shared-only reconstruction. To ensure that *all* structure common to both regions is routed

into the shared block, we define a loss such that each decoder reconstructs the neural activity from its region using *only* the shared representations. Let $\mathbf{w}^{(s)} = \mathbf{w}_1 \odot \mathbf{w}_2 = [\mathbf{1}_{d_s}, \mathbf{0}_{d_1}, \mathbf{0}_{d_2}]^T$ be the intersection mask that selects only the shared dimensions. The loss is

$$\mathscr{L}_{\text{shared}} \;=\; \sum_{r=1}^{2} \big\| D_{\phi}^{(r)}\big((\mathbf{w}^{(s)}\mathbf{1}_T^{\top}) \odot \mathbf{Z}\big) - \mathbf{X}^{(r)} \big\|_F^2, \tag{4.9}$$

where $\mathbf{w}^{(s)}$ zeroes out all private coordinates from the inputs to the decoder, thus forcing the model to encode every cross-region regularity inside the shared subspace. Without this constraint, shared information can shift into private subspaces, leading to inaccurate representation.

Alignment loss. The shared latents are meant to capture only dynamics common to both regions. To enforce this, we align each encoder's shared output to their average, ensuring consistency across regions and preventing region-specific variance from leaking into the shared space:

$$\mathscr{L}_{\text{align}} = \sum_{r=1}^{2} \big\| (\mathbf{w}_r \mathbf{1}_T^{\top}) \odot \mathbf{Z} - (\mathbf{w}_r \mathbf{1}_T^{\top}) \odot \mathbf{Z}^{(r)} \big\|_F^2. \tag{4.10}$$

Orthogonality loss. To encourage each latent coordinate to capture distinct, non-redundant structure, we penalize correlations between *all* rows of the fused latent matrix $\mathbf{Z} \in \mathbb{R}^{D \times T}$. Let $\mathbf{G} = \frac{1}{T}\mathbf{Z}\mathbf{Z}^{\top}$ denote the empirical Gram matrix of latent trajectories, whose off-diagonal entries encode row-wise correlations. We drive only the *off-diagonal* entries of $\mathbf{G}$ towards zero,

$$\mathscr{L}_{\text{orth}} \;=\; \big\| \mathbf{G} - \text{diag}\big(\mathbf{G}\big) \big\|_F^2, \tag{4.11}$$

so that every pair of latent dimensions—shared or private—becomes approximately orthogonal, promoting global disentanglement.

Finally, the complete objective minimizes the weighted sum of the four losses:

$$\mathscr{L} \;=\; \mathscr{L}_{\text{rec}} + \lambda_{\text{align}}\mathscr{L}_{\text{align}} + \lambda_{\text{shared}}\mathscr{L}_{\text{shared}} + \lambda_{\text{orth}}\mathscr{L}_{\text{orth}}. \tag{4.12}$$

Algorithm 4.1. Training Coupled Transformer Autoencoders (CTAE) for two regions

Require: Paired recordings $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$; encoder–decoder parameters θ, ϕ; latent dimensions (d_s, d_1, d_2); loss weights $\lambda_{\text{shared}}, \lambda_{\text{align}}, \lambda_{\text{orth}}$; optimizer $\mathscr{O}$ and learning rate η
Ensure: Trained parameters $\theta^\star, \phi^\star$

Initialize latent masks:
$\mathbf{w}_1 \leftarrow [\mathbf{1}_{d_s}, \mathbf{1}_{d_1}, \mathbf{0}_{d_2}]^\top$
$\mathbf{w}_2 \leftarrow [\mathbf{1}_{d_s}, \mathbf{0}_{d_1}, \mathbf{1}_{d_2}]^\top$
$\mathbf{w}^{(s)} \leftarrow \mathbf{w}_1 \odot \mathbf{w}_2$
while not converged **do**
    Encoder forward pass:
    $\mathbf{Z}^{(1)} \leftarrow E_\theta^{(1)}(\mathbf{X}^{(1)})$
    $\mathbf{Z}^{(2)} \leftarrow E_\theta^{(2)}(\mathbf{X}^{(2)})$
    Latent fusion:
    $\mathbf{Z} \leftarrow$ weighted latent average (Eq. (4.4))
    Extract components:
    $\widehat{\boldsymbol{S}}, \widehat{\boldsymbol{P}}^{(1)}, \widehat{\boldsymbol{P}}^{(2)}$ (Eqs. (4.5), (4.6))
    Region-specific reconstruction:
    $\widehat{\mathbf{X}}^{(1)} \leftarrow D_\phi^{(1)}((\mathbf{w}_1 \mathbf{1}_T^\top) \odot \mathbf{Z})$
    $\widehat{\mathbf{X}}^{(2)} \leftarrow D_\phi^{(2)}((\mathbf{w}_2 \mathbf{1}_T^\top) \odot \mathbf{Z})$
    Compute total loss:
    $\mathscr{L} \leftarrow \mathscr{L}_{\text{rec}} + \lambda_{\text{shared}} \mathscr{L}_{\text{shared}} + \lambda_{\text{align}} \mathscr{L}_{\text{align}} + \lambda_{\text{orth}} \mathscr{L}_{\text{orth}}$
    Update parameters:
    $(\theta, \phi) \leftarrow \mathscr{O}(\theta, \phi, \nabla_{\theta,\phi} \mathscr{L}, \eta)$
end while
return $\theta^\star, \phi^\star$

The weights $\lambda_{\text{shared}}, \lambda_{\text{align}}, \lambda_{\text{orth}}$ are selected on a held-out validation set. Methodology for training the architecture is summarized in Algorithm 4.1.

4.4 Extending CTAE to multi-region recordings

In the main text, we presented CTAE with a two-region formulation and reported results on datasets involving either two or three regions. Here, we provide the generalization of our framework to $R \geq 3$ regions.

4.4.1 Problem formulation for three regions

For clarity, we present the formulation below for three regions. However, this construction naturally extends to more than three regions.

Let $\boldsymbol{X}^{(1)} \in \mathbb{R}^{N_1 \times T}, \boldsymbol{X}^{(2)} \in \mathbb{R}^{N_2 \times T}, \boldsymbol{X}^{(3)} \in \mathbb{R}^{N_3 \times T}$ denote simultaneous neural population recordings from three brain regions, each acquired over T time steps with N_1, N_2, N_3 channels, respectively. We assume that the observed activity in each region arises from latent dynamics that decompose into:

1. **Region-private dynamics $\boldsymbol{P}^{(r)}$** that capture computations unique to each region $r \in \{1,2,3\}$, lying in subspaces orthogonal to shared components.

2. **Pairwise shared dynamics $\boldsymbol{S}^{(ij)}$** that span communication subspaces shared between each pair of regions $(i,j) \in \{(1,2),(2,3),(1,3)\}$, reflecting coordinated activity patterns specific to that pair.

3. **Globally shared dynamics $\boldsymbol{S}^{(123)}$** that capture population trajectories common to all three regions, representing fully shared circuit-level computations.

Formally, the neural activity at each time step t is modeled as nonlinear functions of these latent processes:

$$\begin{aligned}
\mathbf{X}_t^{(1)} &= f\Big(\boldsymbol{S}_{1:t}^{(12)}, \boldsymbol{S}_{1:t}^{(13)}, \boldsymbol{S}_{1:t}^{(123)}, \boldsymbol{P}_{1:t}^{(1)}\Big), \\
\mathbf{X}_t^{(2)} &= g\Big(\boldsymbol{S}_{1:t}^{(12)}, \boldsymbol{S}_{1:t}^{(23)}, \boldsymbol{S}_{1:t}^{(123)}, \boldsymbol{P}_{1:t}^{(2)}\Big), \\
\mathbf{X}_t^{(3)} &= h\Big(\boldsymbol{S}_{1:t}^{(13)}, \boldsymbol{S}_{1:t}^{(23)}, \boldsymbol{S}_{1:t}^{(123)}, \boldsymbol{P}_{1:t}^{(3)}\Big),
\end{aligned} \tag{4.13}$$

where $\boldsymbol{S}_t^{(ij)} \in \mathbb{R}^{d_{ij}}$ denotes the latent state shared between regions i and j, $\boldsymbol{S}_t^{(123)} \in \mathbb{R}^{d_{123}}$ the fully shared latent state across all three regions, and $\boldsymbol{P}_t^{(r)} \in \mathbb{R}^{d_r}$ the region-private latent state of region r.

Our objective is to recover these latent representations $\{\boldsymbol{S}^{(ij)}, \boldsymbol{S}^{(123)}, \boldsymbol{P}^{(r)}\}$ from the observed neural activity $\{\mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \mathbf{X}^{(3)}\}$, such that the model disentangles private, pairwise shared, and fully shared dynamics in a principled way. For more than three regions, this formulation extends by including shared latent variables for all possible combinatorial subsets of

regions, with the training algorithm remaining unchanged. The training algorithm introduced below applies to this general multi-region setting.

4.4.2 General multi-region formulation

To move beyond two regions, we extend the formulation with a general mechanism that assigns latent dimensions to all possible combinatorial subsets of regions. This extension builds directly on the same architectural backbone introduced in Section 4.3, but augments it with an enhanced weighted latent fusion mechanism. This mechanism enables CTAE to disentangle latents shared by arbitrary subsets of regions while preserving strictly region-private structure. All other components—per-region causal Transformer encoders/decoders and the optimization procedure—remain unchanged. The key modifications are described below.

Encoder outputs. For each region $r \in \{1,\dots,R\}$ we keep a dedicated encoder–decoder pair $\left(E_\theta^{(r)}, D_\theta^{(r)}\right)$ and obtain

$$\mathbf{Z}^{(r)} \;=\; E_\theta^{(r)}(\mathbf{X}^{(r)}) \in \mathbb{R}^{D\times T}.$$

Region-specific weight masks. A single binary matrix encodes which latent dimensions are associated with each region *uses*:

$$\mathbf{W} = \left[\mathbf{w}_1\ \mathbf{w}_2\ \dots\ \mathbf{w}_R\right]^\top \in \{0,1\}^{R\times D}, \qquad \mathbf{W}[r,d] = 1 \iff \text{region } r \text{ claims dimension } d.$$

A dimension is *private* when exactly one row in $\mathbf{W}$ is active and *shared* when two or more rows are active, thereby accommodating every subset pattern (for instance, when $R = 3$ the binary code `110` indicates a dimension used by regions 1 and 2 but not 3; `101` corresponds to regions 1 and 3; and `111` denotes a dimension shared by all three). Relative to the two–region mask 4.3, $\mathbf{W}$ expands the notion of "shared" from "both regions" to "arbitary subset of regions". The membership matrix $\mathbf{W}$ can be (i) fixed a priori from anatomical knowledge, (ii) selected by a hyper-parameter search that allocates an appropriate number of latent dimensions to each subset pattern (analogous to the two-region masks in Eq. (4.3)), or (iii) treated as a fully learnable

variable and optimised jointly with the rest of the network.

Weighted latent fusion. We fuse the region–specific latents with a masked average

$$\mathbf{Z}_t[d] \;=\; \frac{\sum_{r=1}^{R} \mathbf{W}[r,d]\,\mathbf{Z}_t^{(r)}[d]}{\sum_{r=1}^{R} \mathbf{W}[r,d]}, \tag{4.14}$$

so that private dimensions stay unchanged, whereas any dimension claimed by multiple regions is *forced* to align across them.

Region-specific decoding. Each decoder receives only the dimensions it owns,

$$\widehat{\mathbf{X}}^{(r)} = D_{\phi}^{(r)}\big((\mathbf{w}_r \mathbf{1}_T^{\top}) \odot \mathbf{Z}\big),$$

exactly as in the two–region case. Hence the extension does not change model capacity but merely broadens how "shared" information is defined.

This design (i) embeds all regions in one latent space of size D, (ii) preserves region-specific structure through fixed relevance masks, and (iii) enforces cross-region consistency via Eq. (4.14).

4.4.3 Multi-region training objective

We retain the four losses used for two regions—reconstruction, shared-only reconstruction, alignment, and orthogonality—while replacing two binary masks with the matrix $\mathbf{W}$.

Reconstruction. The reconstruction loss is unchanged, only we now sum over all regions.

$$\mathscr{L}_{\text{rec}} = \sum_{r=1}^{R} \big\|\widehat{\mathbf{X}}^{(r)} - \mathbf{X}^{(r)}\big\|_F^2. \tag{4.15}$$

Shared-only reconstruction. To guarantee that *all* cross-region regularities live inside shared dimensions, we mask out every coordinate used by fewer than two regions. Let $\mathbf{s} = \mathbf{1}_{\{\sum_r \mathbf{W}_{r,\cdot} \geq 2\}} \in$

$\{0,1\}^D$ and $\mathbf{w}_r^{(s)} = \mathbf{w}_r \odot \mathbf{s}$. We then reconstruct each region from the shared subspace alone:

$$\mathscr{L}_{\text{shared}} = \sum_{r=1}^{R} \left\| D_\phi^{(r)} \left((\mathbf{w}_r^{(s)} \mathbf{1}_T^\top) \odot \mathbf{Z} \right) - \mathbf{X}^{(r)} \right\|_F^2. \tag{4.16}$$

When $R = 2$ this term reduces to Eq. (11) in the main text, but for $R > 2$ it now supervises *every* shared subset simultaneously.

Alignment. Each region's encoder estimate should coincide with the fused latent over the dimensions it owns:

$$\mathscr{L}_{\text{align}} = \sum_{r=1}^{R} \left\| (\mathbf{w}_r \mathbf{1}_T^\top) \odot \mathbf{Z} - (\mathbf{w}_r \mathbf{1}_T^\top) \odot \mathbf{Z}^{(r)} \right\|_F^2. \tag{4.17}$$

Private coordinates contribute zero because they match by definition; shared coordinates are actively driven together, extending the simple two-region difference penalty to an R-way match.

Orthogonality.

$$\mathscr{L}_{\text{orth}} = \left\| \tfrac{1}{T} \mathbf{Z}\mathbf{Z}^\top - \text{diag}\left(\tfrac{1}{T} \mathbf{Z}\mathbf{Z}^\top \right) \right\|_F^2. \tag{4.18}$$

Identical to the two-region version, this term keeps all latent dimensions, private or shared, mutually decorrelated.

Total loss.

$$\mathscr{L} = \mathscr{L}_{\text{rec}} + \lambda_{\text{shared}} \mathscr{L}_{\text{shared}} + \lambda_{\text{align}} \mathscr{L}_{\text{align}} + \lambda_{\text{orth}} \mathscr{L}_{\text{orth}}, \tag{4.19}$$

with $\lambda_{\text{shared}}, \lambda_{\text{align}}, \lambda_{\text{orth}}$ selected on a validation set. The objective reduces exactly to the two-region formulation when $R = 2$; for $R > 2$ it differs only in how masks are constructed and applied, maintaining conceptual continuity while capturing richer patterns of shared neural dynamics.

4.5 Empirical evaluation on controlled toy dataset

We construct a synthetic two-region dataset in which the ground-truth shared and private latent trajectories are known. We generate paired sequences $X^{(1)} \in \mathbb{R}^{N\times T\times D_1}$ and $X^{(2)} \in \mathbb{R}^{N\times T\times D_2}$ from latent trajectories $S \in \mathbb{R}^{N\times T\times d_s}$, $P^{(1)} \in \mathbb{R}^{N\times T\times d_1}$, and $P^{(2)} \in \mathbb{R}^{N\times T\times d_2}$. Unless otherwise stated, we use $N = 1000, \quad T = 32, \quad d_s = 4, \quad d_1 = d_2 = 3, \quad D_1 = D_2 = 32.$ The latent trajectories S_t, $P_t^{(1)}$, and $P_t^{(2)}$ are generated as sinusoidal processes with randomly sampled amplitudes, phases, and integer frequencies. Concretely, for each trial $i \in \{1,\dots,N\}$ and latent dimension k, we sample an amplitude $a_{i,k} \sim \mathcal{N}(0,1)$, a phase $\phi_{i,k} \sim \text{Uniform}(0,2\pi)$, and a frequency $f_{i,k} \sim \text{Uniform}\{2,3,4\}$, and define

$$z_{i,t,k} \;=\; a_{i,k}\,\sin\big(2\pi f_{i,k}\,\tau_t + \phi_{i,k}\big), \qquad \tau_t = \frac{t}{T},\;\; t = 1,\dots,T, \tag{4.20}$$

The shared latents S and private dimensions $P^{(1)}$ and $P^{(2)}$ are generated independently using the above construction with dimensionality d_s, d_1 and d_2, respectively. Each region linearly mixes shared and private latents:

$$H_t^{(1)} = \alpha_s A_s^{(1)} S_t + \alpha_p A_p^{(1)} P_t^{(1)}, \qquad H_t^{(2)} = \alpha_s A_s^{(2)} S_t + \alpha_p A_p^{(2)} P_t^{(2)},$$

where $A_s^{(r)} \in \mathbb{R}^{D_r\times d_s}$ and $A_p^{(r)} \in \mathbb{R}^{D_r\times d_r}$ have i.i.d. Gaussian entries scaled by $1/\sqrt{d}$. We set $\alpha_s = \alpha_p = 2.0$.

We then introduce nonlinear distortion by applying an elementwise mapping:

$$X_{t,\text{clean}}^{(r)} = \tanh\big(H_t^{(r)}\big) + 0.1\big(H_t^{(r)}\big)^2$$

The resulting $X_{t,\text{clean}}^{(r)}$ can be interpreted as synthetic *log-firing rates*. We then add independent Gaussian noise:

$$X_t^{(r)} = X_{t,\text{clean}}^{(r)} + \varepsilon_t^{(r)}, \qquad \varepsilon_t^{(r)} \sim \mathcal{N}(0,\sigma^2 I),$$

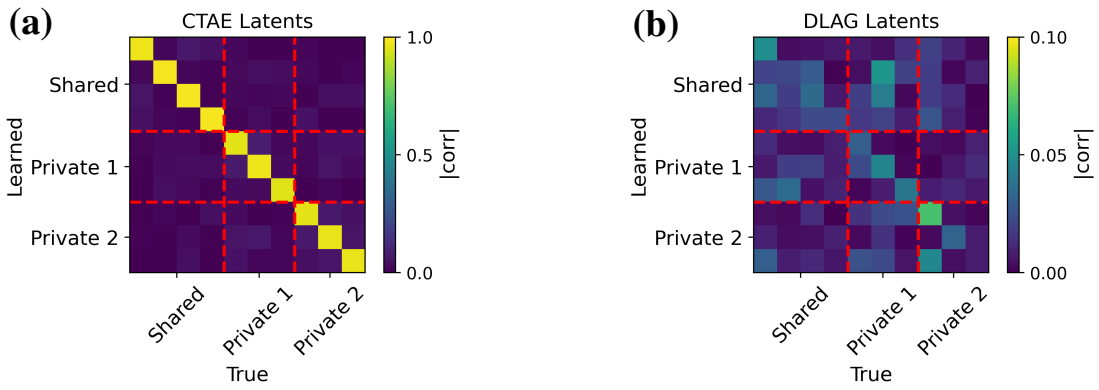


Figure 4.3. Absolute Pearson correlation between aligned learned latents and ground-truth latents on the synthetic dataset. (a) CTAE. (b) DLAG. Red dashed lines indicate shared and region-specific partitions.

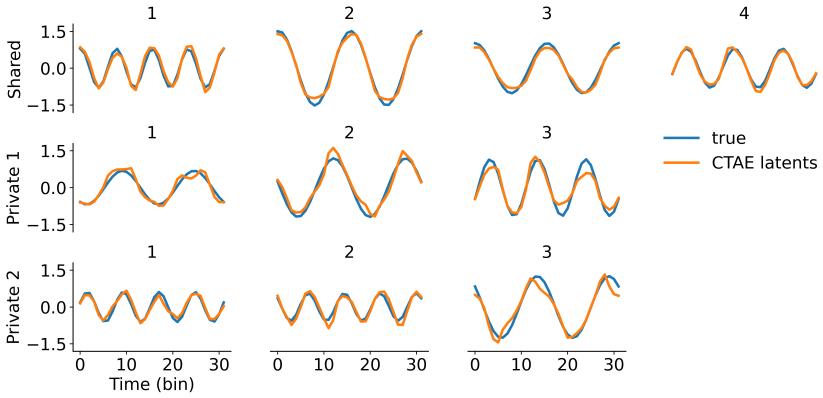


Figure 4.4. Latent trajectories on the synthetic dataset. Rows correspond to shared, private region 1, and private region 2 latents. Blue: ground truth. Orange: aligned CTAE latents.

with $\sigma = 0.01$.

After training, we extract the learned latent trajectories from each model and align each latent subspace (shared and private) to the corresponding ground-truth latents using a linear transformation estimated by least-squares regression on a held-out subset of trials. This alignment accounts for the fact that latent representations are only defined up to a linear transformation.

We then compute the absolute Pearson correlation between each aligned learned latent dimension and each true latent dimension. Figure 4.3 shows the resulting correlation matrices. For CTAE, the correlations exhibit a clear block structure: dimensions assigned to the shared subspace align with the true shared latents, and region-specific dimensions align with their corre-

sponding private latents, with minimal or no cross-block correlation. In contrast, DLAG does not produce a structured alignment; correlations are weak and do not follow the shared/private partition. DLAG assumes a linear observation model, whereas in this dataset the observations are nonlinear functions of the latent variables. This model mismatch limits its ability to recover the underlying shared and private structure. Figure 4.4 further illustrates representative latent trajectories after alignment. CTAE closely matches the ground-truth shared and private dynamics across time, while preserving separation between shared and region-specific components. Overall, CTAE recovers latent dimensions that closely match the true shared and private dynamics under nonlinear observations, whereas DLAG fails to recover the underlying structure in this setting.

4.6 Applications to motor cortical dynamics: M1–PMd

We first evaluate CTAE on a canonical motor circuit dataset in which coordinated population activity across cortical areas supports both movement planning and execution. Specifically, we analyze simultaneous recordings from dorsal premotor cortex (PMd) and primary motor cortex (M1) during a delayed center-out reaching task [Perich et al., 2018]. This dataset provides an ideal testbed for assessing whether CTAE can (i) disentangle shared inter-regional dynamics from region-specific activity, and (ii) recover latent representations that support accurate decoding of both continuous kinematics and discrete movement goals.

4.6.1 Dataset and behavioral task

We use the dataset from [Perich et al., 2018], which includes neuronal recordings from a Utah Array in M1 and PMd (see Fig. 4.5a). In this task, a monkey performs center-out arm reaches using a planar manipulandum. Neural activity was recorded from the primary motor cortex (M1) and dorsal premotor cortex (PMd) via Utah arrays, yielding spike-sorted activity from 52 putative units in M1 and 66 in PMd (Fig. 4.5b). Each trial begins with a target onset, followed by a variable delay period (0.5–1.3 s) for movement preparation. After a "go" cue,

the monkey initiates a reach to one of eight evenly spaced targets in a 2D workspace, holds briefly, and returns to center. The dataset includes 208 trials across 8 target conditions, with approximately 26 repetitions per condition.

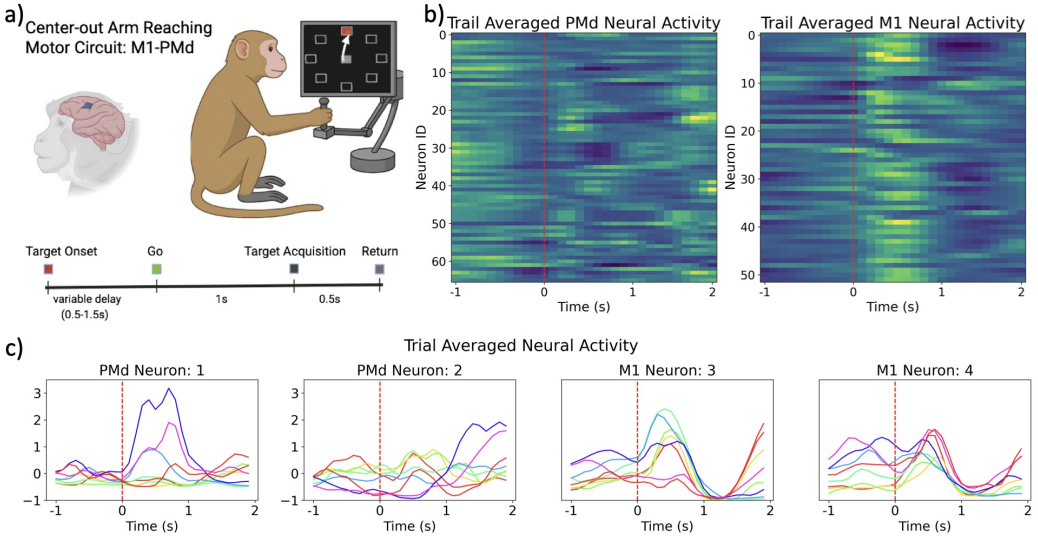


Figure 4.5. Motor study. a) Schematic of the behavioral tasks performed by non-human primates in the datasets used in Section 2.4. Center-Out Arm Reach Task: The monkey controls a manipulandum to perform instructed delay reaches toward one of eight peripheral targets, with neural activity recorded from M1 and PMd using Utah arrays. b) Population firing rates in M1 and PMd. Heatmaps show the trial-averaged z-scored firing rates for individual neurons (rows) across time. c) Condition-averaged firing rates of individual neurons in M1 and PMd. Each panel shows trial-averaged firing rates for single neurons, grouped and colored by reach direction. Colormap follows that in Fig. 4.8.

Historically, PMd and M1 have been jointly analyzed due to their complementary roles in movement preparation and execution. During the instructed delay period between stimulus onset and the go cue, both regions exhibit preparatory activity without inducing muscle output [Kaufman et al., 2014, Churchland and Shenoy, 2024], potentially representing task goals—i.e., target identity [Byron et al., 2010, Lara et al., 2018]. Following the go cue, both PMd and M1 generate execution-related activity that descends to motor neurons and enables decoding of continuous hand kinematics, including position and velocity [Churchland and Shenoy, 2024, Elsayed et al., 2016, Gilja et al., 2012, Byron et al., 2010]. These established properties motivate testing whether CTAE can isolate shared PMd–M1 task-relevant dynamics while preserving

region-private structure.

4.6.2 CTAE latent dynamics in the motor circuit

We applied CTAE to the simultaneous PMd–M1 recordings to obtain (i) a shared subspace of dimension d_s, capturing correlated dynamics across PMd and M1, and (ii) private subspaces of dimensions d_{PMd} and d_{M1}, capturing region-specific activity. The inferred shared and region-private latents are shown in Fig. 4.6. Each panel corresponds to a single latent dimension evolving over time, with colors denoting trial-averaged activity across reach directions. Distinct latent dimensions exhibit systematic variation across reach directions, indicating that the learned representations capture behaviorally relevant structure. The shared latent subspace reflects both condition-invariant and condition-dependent dynamics, encoding aspects of reach direction and temporal progression, whereas region-specific latents display more complex dynamics consistent with local circuit computations.

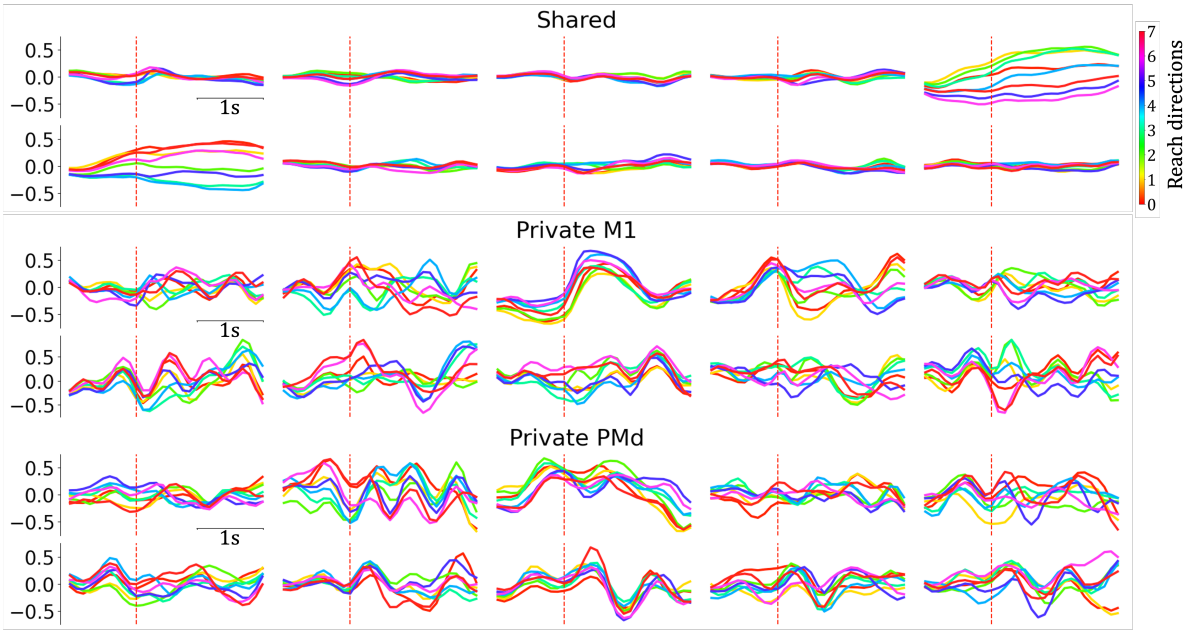


Figure 4.6. CTAE latent dynamics in M1–PMd. Each subplot visualizes a single neural latent, with colors indicating trial-averaged latents for each reach direction. Red dashed line indicates go cue. (Top) Shared latents capture neural dynamics present in both M1 and PMd. (Middle) M1-private latents capture dynamics specific to M1. (Bottom) PMd-private latents capture dynamics specific to PMd.

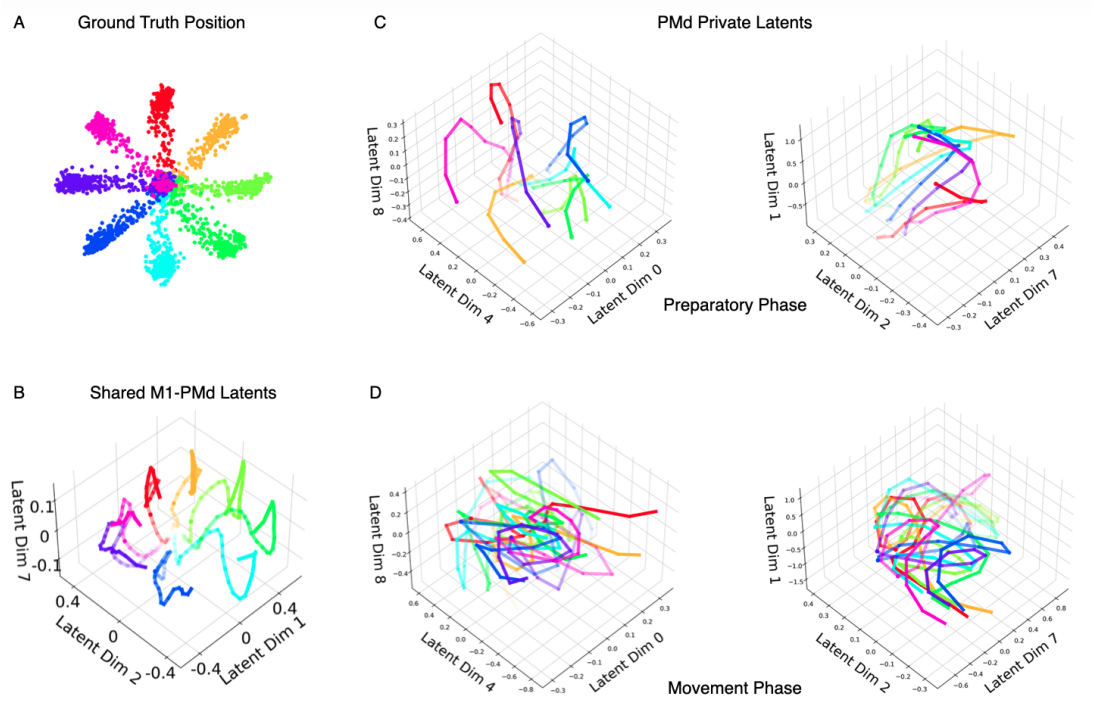


Figure 4.7. State-space views of CTAE latents across task epochs. Each subplot shows trajectories in a 3D latent subspace. Colors denote trial-averaged trajectories for each reach direction in (a). (b) Shared PMd–M1 subspace: trajectories evolve over the entire trial; circle denotes go cue. (c) PMd-private subspace during the preparatory phase (two different 3D projections shown). (d) PMd-private subspace during the movement phase (same projections as in (c)). Together, these panels highlight distinct PMd–M1 interaction modes that differ between preparation and execution.

To further examine the geometry of these representations, Fig. 4.7 visualizes state-space trajectories of selected latent groups across task epochs. Immediately after target onset, trajectories in the shared subspace $Z^{(s)}$ diverge from a common baseline toward target-dependent fixed points, forming a goal-related preparatory representation consistent with prior reports of a PMd–M1 preparatory subspace. Following the go cue, variance in $Z^{(s)}$ expands along axes orthogonal to the preparatory manifold, reflecting a shift in PMd–M1 coupling as the circuit transitions from planning to execution (Fig. 4.7b). In contrast, PMd-private latents maintain a stable target representation during the preparatory phase (Fig. 4.7c) and weaken after movement initiation (Fig. 4.7d), consistent with transient reorganization as movement unfolds. Together, these analyses highlight distinct interaction modes across preparation and execution that align

with established views of motor-circuit population dynamics.

4.6.3 Behavioral decoding from CTAE latent subspaces

To assess the behavioral relevance of the inferred latent representations, we asked whether behaviorally relevant information is captured by the shared or by the region-private subspaces recovered by CTAE. In particular, we evaluated whether different aspects of behavior—continuous hand kinematics and discrete reach goals—could be decoded from these subspaces using simple linear readouts. By comparing decoding performance across shared and private latents, this analysis reveals how task-related information is distributed across the PMd–M1 circuit and whether the shared communication subspace concentrates the dominant behaviorally relevant signals.

For continuous decoding, we predicted hand position at each time point after the go cue (the final two seconds of each trial) using linear regression,

$$v_t^{\text{hand}} = W\mathbf{z}_t,$$

where $\mathbf{z}_t$ denotes the latent representation at time t. For discrete decoding, we classified the reach target (one of eight directions) using multi-class logistic regression,

$$\hat{y}_i = \frac{\exp(\mathbf{w}_i^\top \mathbf{z} + b_i)}{\sum_{j=1}^{C} \exp(\mathbf{w}_j^\top \mathbf{z} + b_j)}, \qquad i = 1, \ldots, C.$$

Figure 4.8 compares decoding performance obtained from raw neural activity, CTAE latents, and DLAG latents.

Decoding results reveal that CTAE concentrates the majority of behaviorally relevant variance within the shared PMd–M1 subspace. As shown in Fig. 4.8a, linear decoding from shared latents accurately reconstructs hand trajectories, indicating that the dominant motor signals underlying movement execution are captured within this shared representation. In

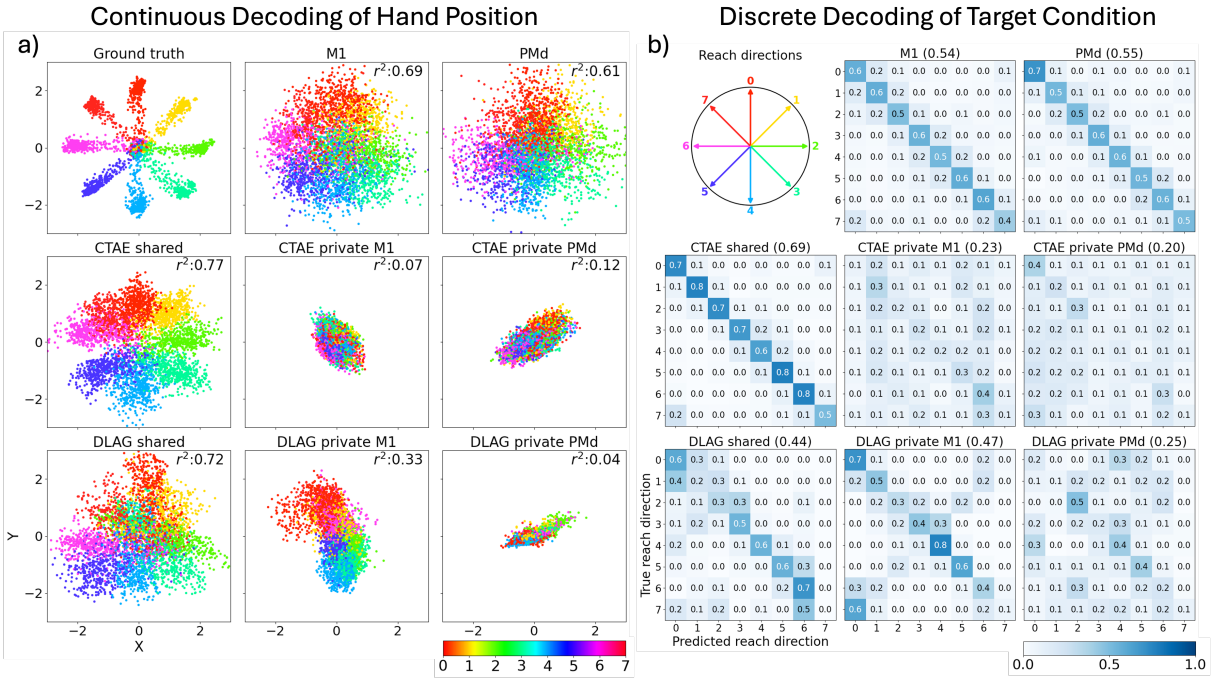


Figure 4.8. M1–PMd dataset. a) Ground truth hand position in top-left corner. Hand position decoding from neural activity in M1 and PMd (top), from CTAE latents (middle), and from DLAG (bottom). b) Confusion matrices for reach direction classification (order of plots is same as in a). Classification accuracy in parentheses.

contrast, DLAG distributes predictive information across private PMd and M1 latents in a directionally anisotropic manner. This fragmentation is particularly evident in the discrete classification task, where DLAG private latents achieve high accuracy only for a subset of reach directions (Fig. 4.8b). Such direction-specific partitioning of kinematic information is inconsistent with established observations that both PMd and M1 encode reach kinematics through a coherent low-dimensional population manifold. The results therefore suggest that DLAG can inadvertently route shared task-relevant variance into private channels, whereas the structured mask design and regularization constraints of CTAE prevent such leakage and promote a coherent shared representation. We additionally compare against DeepCCA and a concatenated-data Transformer autoencoder baseline (Appendix 4.6.5), both of which achieve substantially lower decoding accuracy than CTAE shared latents.

To further characterize how task information evolves during the trial, we performed

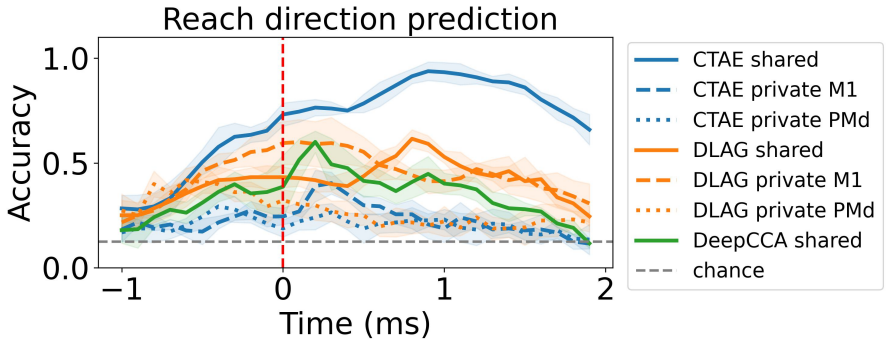


Figure 4.9. Time-resolved reach-direction decoding accuracy (5-fold CV) for each subspace across multiple models. Red dashed line represents the go cue. Shared latents from all models show a sharp increase in accuracy following the go cue, with CTAE shared latents consistently outperforming others. PMd-specific latents exhibit a pre-go-cue rise, indicating anticipatory directional information, whereas M1-specific latents peak post go cue, reflecting direction-specific encoding after movement initiation. Shading represents ± 1 standard error across folds.

time-resolved decoding of reach direction across the entire trial duration (Fig. 4.9). Directional information rises sharply following the go cue across all models, reflecting the transition from movement preparation to execution. However, CTAE shared latents consistently achieve the highest decoding accuracy throughout the movement epoch. Notably, PMd-private latents show elevated decoding accuracy prior to the go cue, suggesting that PMd carries anticipatory directional information during the preparatory phase. In contrast, M1-private latents reach peak accuracy after movement initiation, consistent with stronger representation of movement execution in M1. These temporal dynamics align with established views of motor cortical function, in which PMd encodes preparatory signals related to upcoming movement goals while M1 more strongly reflects movement execution [Lara et al., 2018].

4.6.4 Ablation study: contribution of individual loss terms

To assess which components of the CTAE objective are necessary to obtain a predictive and interpretable shared–private decomposition, we performed an ablation study on the M1–PMd dataset. We trained four variants: (i) the full CTAE objective; (ii) without the shared-only reconstruction term; (iii) without the alignment term; and (iv) without the orthogonality

constraint. We quantify downstream utility by reach-direction decoding accuracy from each subspace (shared, private M1, private PMd), reported in Table 4.1. In addition, we visualize representative qualitative failure modes for the alignment and orthogonality ablations (Figs. 4.10–4.11).

Shared-only reconstruction loss.

The shared-only reconstruction term enforces that the shared subspace alone can reconstruct each region's activity, thereby encouraging shared latents to capture *all* common task-relevant dynamics rather than allowing shared information to leak into private dimensions. Removing this loss produces a pronounced collapse in shared-latent decoding accuracy (Table 4.1), accompanied by a substantial increase in private-latent accuracies. This tradeoff indicates that, without explicit pressure for the shared subspace to be sufficient for reconstruction, the model can route predictive information into private channels, weakening interpretability of the shared representation.

Alignment loss.

The alignment loss penalizes discrepancies between the shared latents inferred from each region, ensuring that the two encoders identify a consistent shared trajectory rather than two unrelated representations that happen to reconstruct well. Qualitatively, when alignment is enforced, shared latents from PMd and M1 closely overlap (Fig. 4.10a). When alignment is removed, corresponding shared dimensions can diverge substantially and sometimes partially cancel when fused (Fig. 4.10b), indicating a breakdown in cross-region correspondence. Quantitatively, this degradation is reflected by reduced decoding accuracy from the shared subspace (Table 4.1).

Orthogonality loss.

The orthogonality constraint encourages disentanglement by minimizing redundancy between shared and private representations. When this loss is relaxed, shared and private subspaces become less separable, leading to substantial overlap as measured by latent dot products (Fig. 4.11b). In this regime, predictive information is no longer concentrated in a clean

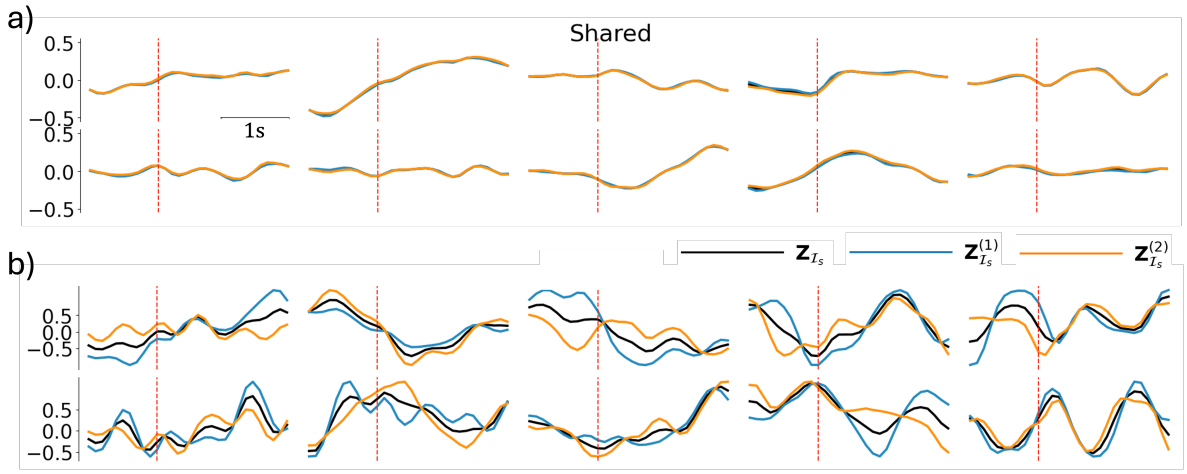


Figure 4.10. Alignment of shared latents in M1–PMd. Each subplot shows a representative shared latent. Blue and orange traces correspond to shared latents inferred from M1 and PMd, respectively; the black trace denotes their fused mean. (a) With alignment loss enforced, shared latents exhibit strong overlap across regions. (b) Without alignment, shared latents diverge substantially, indicating degraded cross-region correspondence.

shared component, and shared-latent decoding accuracy drops sharply (Table 4.1), while private latents become spuriously more predictive.

Together, these ablations show that all three terms play complementary roles: shared-only reconstruction concentrates common task-relevant variance into the shared subspace; alignment enforces cross-region consistency of shared trajectories; and orthogonality prevents redundancy and leakage between shared and private channels. Removing any one of these terms degrades either predictive performance, interpretability, or both (Table 4.1).

Table 4.1. Ablation study of CTAE loss functions. Each entry indicates the model's accuracy in predicting the reach direction. Values in parentheses indicate standard deviation across 5 folds.

Model	**Shared**	**Private M1**	**Private PMd**
CTAE	0.69 (0.03)	0.22 (0.02)	0.21 (0.03)
CTAE without alignment loss	0.61 (0.02)	0.16 (0.02)	0.2 (0.02)
CTAE without orthogonality loss	0.31 (0.02)	0.28 (0.02)	0.29 (0.02)
CTAE without shared only reconstruction loss	0.34 (0.01)	0.36 (0.01)	0.37 (0.01)

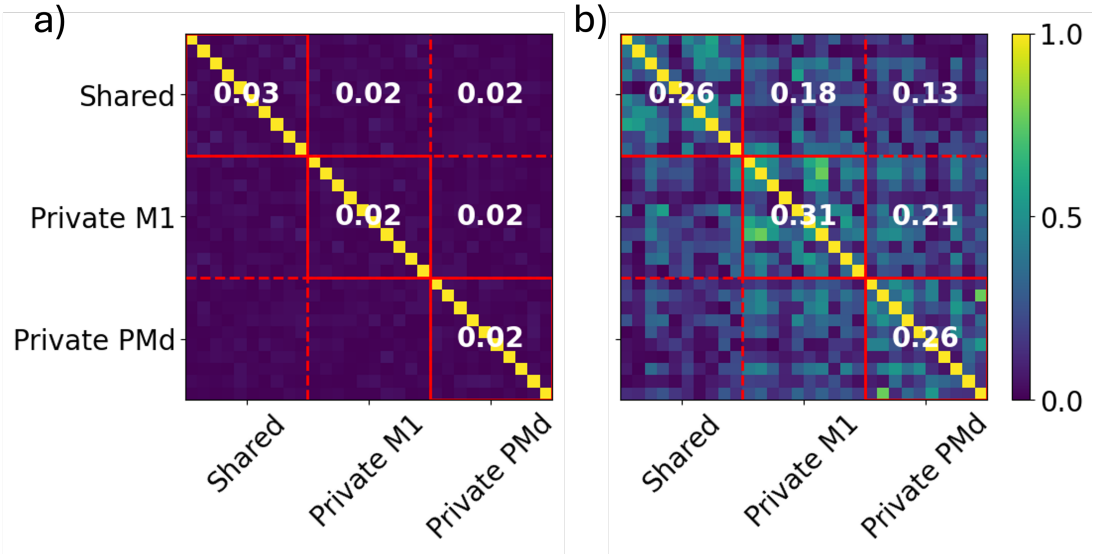


Figure 4.11. Orthogonality between CTAE latents in M1–PMd. Blocks show mean dot products between latent dimensions; red separators delineate shared, M1-private, and PMd-private groups. (a) With orthogonality enforced, shared and private latents remain well separated. (b) Without orthogonality, substantial overlap emerges, indicating loss of disentanglement.

4.6.5 Baseline comparisons

Comparison to the concatenated-data baseline

As a baseline, we applied the same transformer autoencoder backbone to the concatenation of neural activity from M1 and PMd, without explicitly separating latent dimensions into shared and region-private subspaces. In this setting, the model is trained using only the reconstruction loss (Eq. 4.8, applied jointly across regions) and the orthogonality regularization (Eq. 4.11), but without alignment or shared-only reconstruction constraints. This baseline provides a reference for assessing how much task-relevant information can be extracted when multi-region data are modeled jointly, but without enforcing a structured decomposition into shared and private components.

The concatenated-data model was evaluated on the same continuous and discrete decoding tasks used throughout this section. On continuous hand-position decoding, the concatenated baseline achieved an R^2 of 0.59, compared to 0.77 for CTAE shared latents and 0.72 for DLAG shared latents (Fig. 4.12a). For discrete reach-direction classification, the concatenated model

attained an accuracy of 57%, substantially lower than CTAE (69%) and also below DLAG (44%) when evaluated on shared representations (Fig. 4.12b).

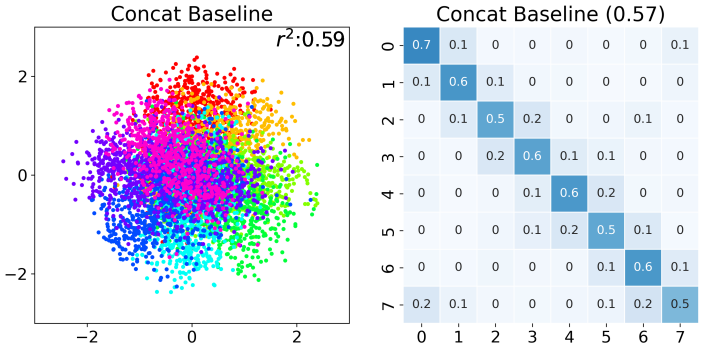


Figure 4.12. Decoding performance of the concatenated-data baseline on the M1–PMd dataset. (a) Scatter plot of predicted versus true hand positions obtained from linear decoding of concatenated latents. (b) Confusion matrix for single-trial reach-direction classification. True and predicted reach conditions are shown on the Y- and X-axes, respectively.

These results demonstrate that simply pooling activity from multiple regions into a single latent space is insufficient to recover the full structure of task-relevant dynamics. In contrast, explicitly separating shared and region-private components,as enforced by CTAE, concentrates behaviorally relevant information into a coherent shared subspace while preserving region-specific variability, leading to superior decoding performance.

Comparison to DeepCCA

To further contextualize the performance of CTAE, we compared it to Deep Canonical Correlation Analysis (DeepCCA), a widely used nonlinear method for extracting shared representations across paired datasets. DeepCCA aims to learn nonlinear transformations of each region's activity such that the resulting latent variables are maximally correlated, but it does not explicitly model temporal structure, shared–private separation, or reconstruction fidelity.

We implemented DeepCCA using parallel multilayer perceptrons (MLPs) for M1 and PMd. Each branch consisted of three fully connected layers with 1024 hidden units, sigmoid nonlinearities, and batch normalization (non-affine) applied after each layer. The final layer projected each branch into a shared latent space matched in dimensionality to the CTAE shared

subspace. Canonical correlation analysis was then applied to maximize correlation between the two branches. The learned shared latents were evaluated on the same downstream decoding tasks used throughout this section.

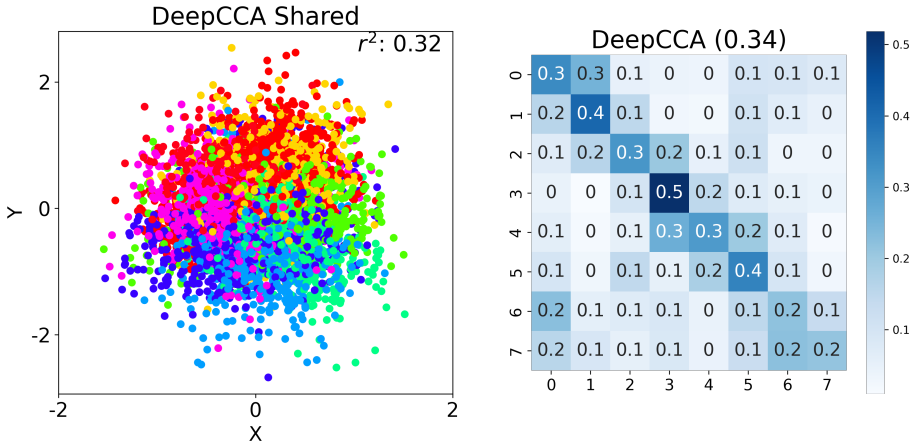


Figure 4.13. Decoding performance of DeepCCA on the M1–PMd dataset. (a) Predicted versus true hand positions obtained from linear decoding of DeepCCA shared latents. (b) Confusion matrix for single-trial reach-direction classification using DeepCCA shared latents.

Continuous decoding of hand position, using linear regression, from DeepCCA shared latents yielded an R^2 value of 0.32 (Fig. 4.13a), substantially lower than the performance obtained from CTAE shared latents ($R^2 = 0.77$) and DLAG shared latents ($R^2 = 0.72$). This gap reflects DeepCCA's inability to capture temporally structured dynamics when operating on time points treated as independent samples.

For discrete decoding of reach-direction classification, DeepCCA achieved an accuracy of 34% at the single time-point level (Fig. 4.13b), compared to 69% for CTAE and 44% for DLAG when decoding from shared representations. This reduced performance suggests that maximizing instantaneous correlation alone is insufficient to recover task-relevant shared structure in distributed motor circuits.

Overall, these results highlight the limitations of correlation-based multiview methods when applied to temporally extended neural data. While DeepCCA can identify correlated dimensions across regions, it lacks mechanisms to enforce temporal continuity, disentangle

shared from region-specific activity, or preserve information necessary for accurate behavioral decoding. In contrast, CTAE integrates temporal modeling, reconstruction constraints, and structured disentanglement, yielding shared representations that are both more interpretable and substantially more predictive of behavior.

Overall, CTAE identifies a shared PMd–M1 subspace that concentrates the dominant task-relevant motor signals while preserving meaningful region-private dynamics. The shared latents support accurate continuous kinematic decoding and robust reach-direction classification (Fig. 4.8), and time-resolved decoding reveals distinct preparatory versus execution-phase contributions across subspaces (Fig. 4.9). Comparisons to correlation-based (DeepCCA) and unstructured (concatenated-data) baselines show that explicitly modeling temporal dynamics and enforcing a shared–private decomposition is critical for recovering predictive and interpretable shared representations. Finally, ablations confirm that shared-only reconstruction, alignment, and orthogonality constraints are each necessary to prevent leakage and recover a stable shared–private factorization (Table 4.1).

4.7 Applications to multisensory circuit dynamics: SC–ALM

We next evaluate CTAE on a multisensory cortico–subcortical circuit in which distributed population dynamics support evidence integration and decision-making. In contrast to the M1–PMd reaching task (Sec. 4.6), this dataset involves temporally extended sensory evidence with distinct stimulus modalities and a delayed choice, allowing us to probe how shared and region-private subspaces capture stimulus- and choice-related information across circuit components.

4.7.1 Dataset and multisensory task

The anterolateral motor cortex (ALM) and the superior colliculus (SC) are important parts of a cortico-subcortical loop that transforms multisensory inputs into behavioral decisions. To study the role of these areas in multisensory integration and decision-making, we trained

mice in a multisensory discrimination task, where animals had to integrate visual and tactile information over time to identify the target stimulus side. We then performed simultaneous neural recordings in ALM and SC using high-density Neuropixels probes in task-performing animals (Fig. 4.14, left).

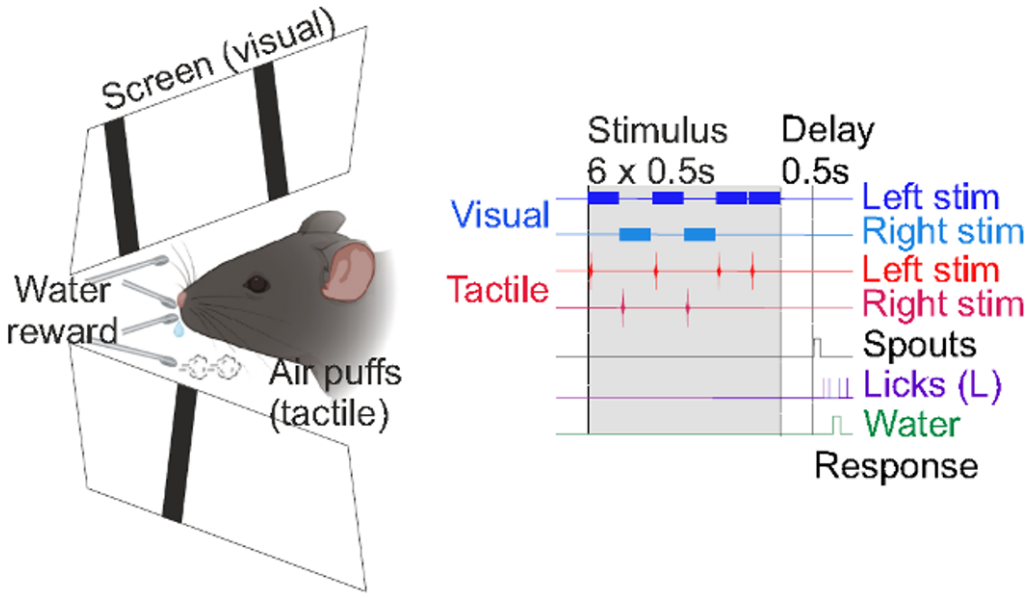


Figure 4.14. Multisensory study. Left: schematic of the 2-alternative forced-choice task performed by mice, trained to lick on the side where more stimuli were presented; correct choices are rewarded at the target spout. Right: schematic of the task timeline in a multisensory trial with a left-side target. The stimulus phase is shown in grey; the delay and response phases are shown in white and separated by a vertical line; spout movement begins at the start of the response period.

Each trial had a total duration of 7 seconds, comprising a 1-second baseline, then 3 seconds of probabilistic stimuli, followed by a 0.5-second delay without any stimulus and finally, the spouts moved in front of the mouse, allowing up to 2.5 seconds for a licking response. The mice were rewarded when responding on the side where more sensory stimuli were presented during the stimulus period. The stimulus period was composed of six periods of 500 milliseconds each, during which a stimulus could be presented to the mouse. One to six stimuli were presented on a given side, with each stimulus likelihood of presentation determining the difficulty of the task. Depending on the trial type, a stimulus could either be a *visual* grating moving along

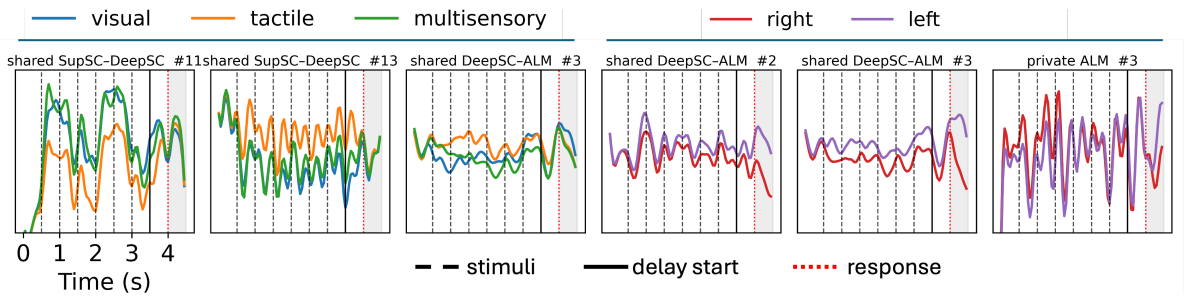


Figure 4.15. SC–ALM dataset. Representative latents. Each panel shows the condition-averaged time course of one latent (title indicates subspace and latent index). Left panels: stimulus modality—visual (blue), tactile (orange), multisensory (green). Right panels: target side—right (red), left (purple).

a screen (visual trial), an *air puff* of 100ms to the vibrissae on one side of the mouse (tactile trial), or both, in a simultaneous congruent *multisensory* manner (multisensory trial). Mice were trained to respond by licking the side on which more stimuli were shown. An example trial is illustrated in Fig. 4.14 (center), and an example multisensory trial timeline is shown in Fig. 4.14 (right).

We analyzed a single session from an expert animal that achieves $\sim 70\%$ correct choice accuracy (564 trials in total), with approximately 80–100 trials per stimulus type and per target direction.

4.7.2 Multi-region CTAE representative latents

We evaluated CTAE on simultaneous recordings from the superior colliculus (SC) and anterolateral motor cortex (ALM) in mice trained on the multisensory discrimination task described in Sec. 4.7.1. Neural activity was recorded from superficial and deep layers of SC together with ALM while animals integrated visual and tactile stimuli to identify the target side (left vs. right). To model this circuit, we fit the multi-region CTAE variant ($R > 2$) to three regions (superficial SC, deep SC, and ALM), enabling identification of shared-across-all, pairwise-shared, and region-private subspaces. Fig. 4.15 shows representative latent trajectories averaged by stimulus modality and target side, illustrating that distinct subspaces capture complementary structure across the trial.

SC and ALM are key components of a cortico-subcortical loop implicated in linking sensory inputs to motor planning. SC integrates visual and tactile information and contributes to orienting behavior [Stein and Meredith, 1993, Cang and Feldheim, 2013], whereas ALM encodes preparatory and choice-related activity during decision-making tasks [Guo et al., 2014, Li et al., 2015]. We hypothesized that shared subspaces across superficial and deep SC layers and between deep SC and ALM would preferentially capture task-relevant features, such as stimulus type and target side, thereby mediating the transformation from multisensory evidence accumulation to decision making.

4.7.3 Behavioral decoding from CTAE latent subspaces

We next evaluated how task-relevant information is represented across the latent subspaces identified by CTAE. Specifically, we examined the ability to decode both stimulus modality and target side from neural activity and from the learned latent representations using the same linear decoding protocol introduced for the M1–PMd analysis (Sec. 4.6.3). Fig. 4.16 summarizes decoding performance across raw neural activity, region-private latent subspaces, and shared latent subspaces.

For the raw neural activity, Deep SC provides the strongest discrimination of both stimulus modality and target side. Superficial SC activity correctly distinguishes target side but shows weaker discrimination between stimulus types for visual and multisensory trials, and also exhibits limited separation between target sides in tactile trials. In contrast, ALM activity reliably distinguishes target side but shows weaker discrimination between stimulus types for tactile and multisensory trials and poorer separation between target sides during visual trials. These observations suggest that individual regions contain complementary but incomplete representations of task variables.

Applying CTAE reveals shared dynamics between regions that better capture stimulus and target information. In particular, the shared subspaces between superficial and deep SC and between deep SC and ALM achieve higher decoding accuracy than the region-private latents. In

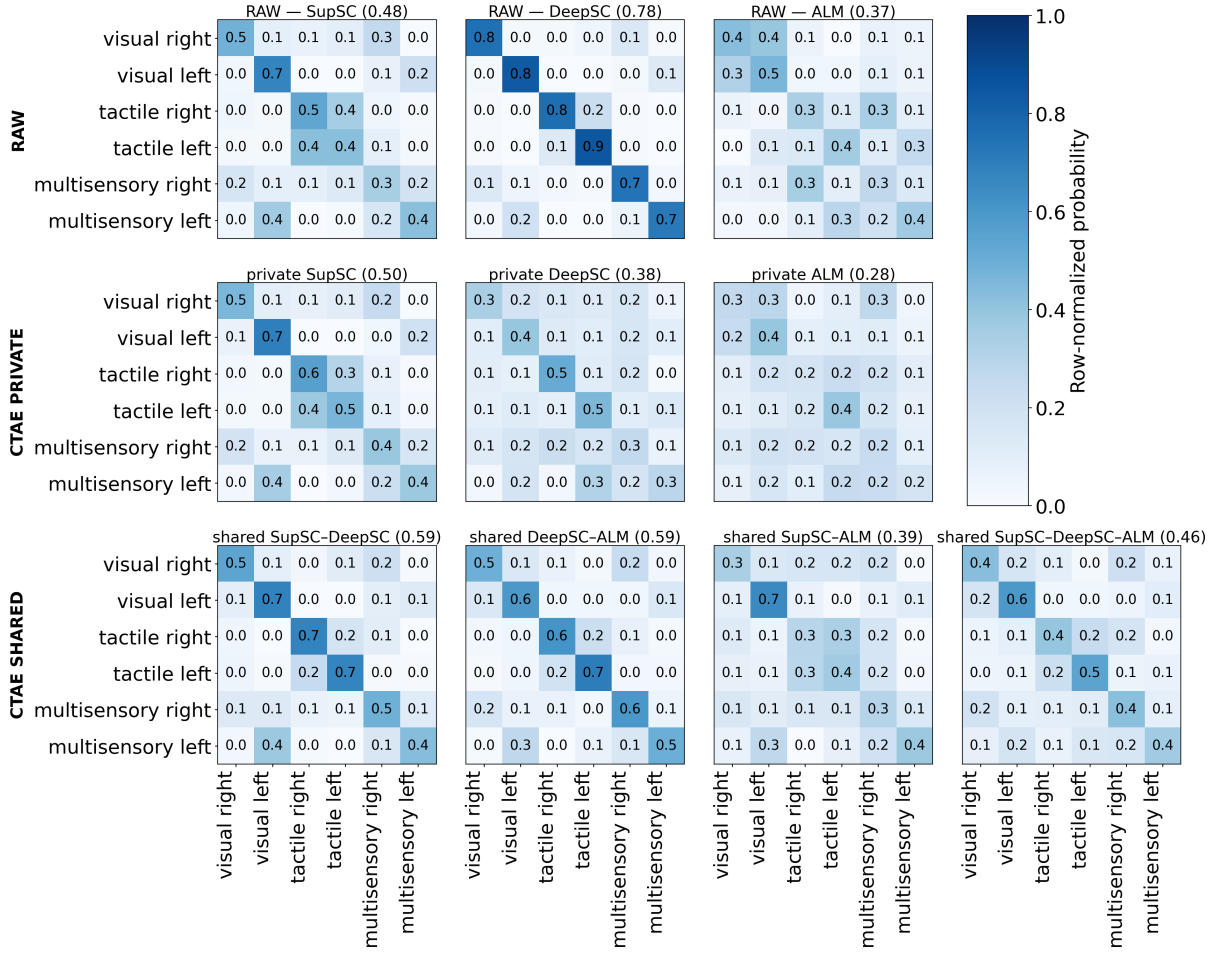


Figure 4.16. Row-normalized confusion matrices for decoding stimulus × target side. (Top) Raw activity from SupSC, DeepSC, and ALM. (Middle) CTAE private (region-specific) latents. (Bottom) CTAE shared latents (pairwise and 3-way). Panel titles show mean 5-fold accuracy.

contrast, the shared subspace between superficial SC and ALM shows little shared dynamics that are useful for discrimination. Together, these results suggest that deep SC plays a central role in the circuit, mediating interactions that link sensory evidence accumulation with decision-related activity, with shared dynamics reflecting evidence integration with superficial SC and choice-related representations with ALM.

For comparison, we applied mDLAG to the same dataset (Fig. 4.17). In contrast to CTAE, mDLAG shared and private latents do not exhibit improvement over decoding from raw neural activity, and decoding performance remains diffuse across latent subspaces. This likely

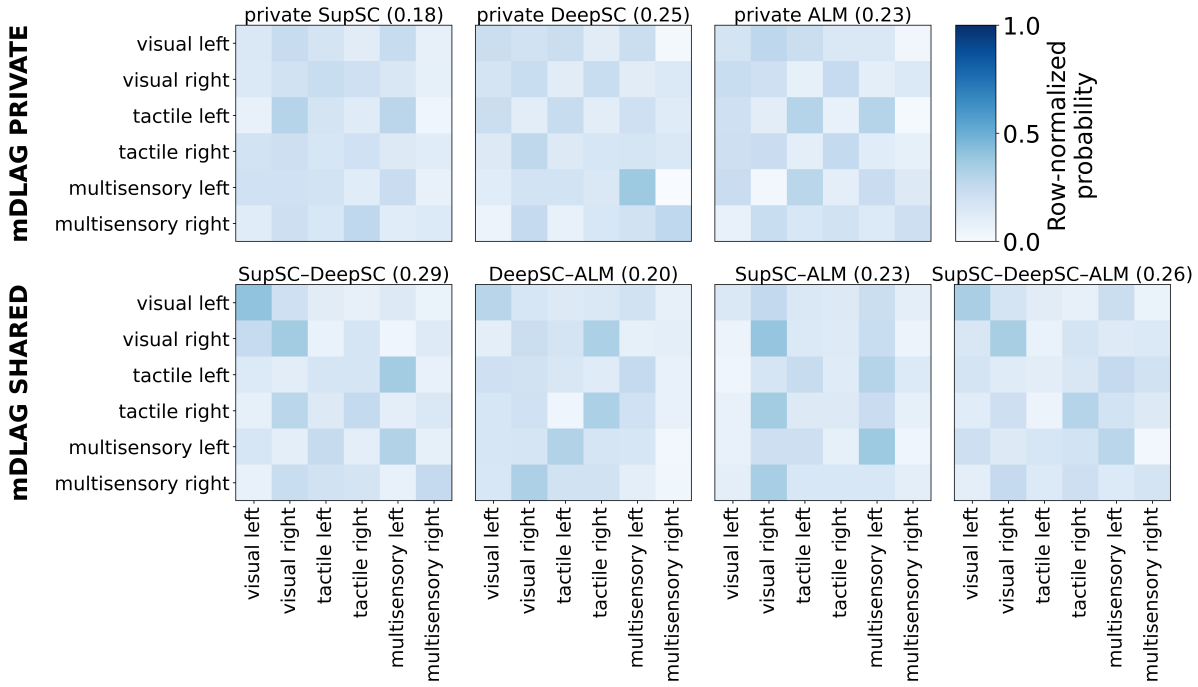


Figure 4.17. Row-normalized confusion matrices for decoding stimulus × target side. (Top) mDLAG private: region-specific latents; (bottom) mDLAG shared latents (pairwise and 3-way). Panel titles show mean 5-fold accuracy.

reflects the linear observation model underlying mDLAG, which may limit its ability to capture nonlinear multi-region interactions present in this multisensory task.

To further characterize how task information evolves over time within the learned representations, we performed time-resolved decoding of the target variable from the CTAE latent subspaces. For this analysis, we used per-trial, per-time-bin latent representations from both region-private and shared subspaces (including pairwise and all-region shared components). Following the same approach used for time-resolved decoding in the motor circuit (Sec. 4.6.3), we trained a single multinomial logistic regression classifier with feature standardization using 5-fold stratified cross-validation on the full time-flattened feature matrix for each subspace. Time-resolved decoding accuracy was then obtained without re-training the classifier: at test time, we fixed the classifier and masked all features outside a temporal context centered at each time point using a symmetric sliding window of five bins (± 2 bins). We report the mean and standard deviation across folds at each time bin and overlay task event markers including

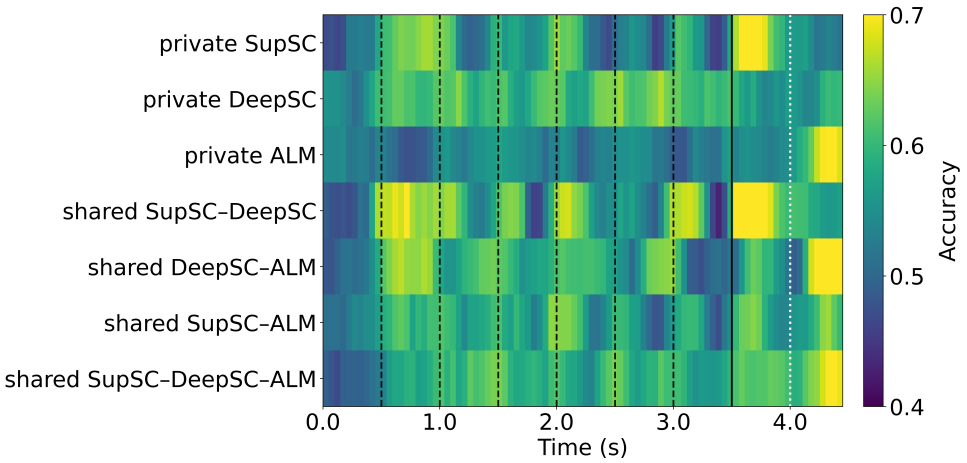


Figure 4.18. Time-resolved decoding accuracy (mean over 5-fold CV) for each latent subspace. Rows correspond to private and shared subspaces (private SupSC/DeepSC/ALM; shared pairs and all-shared), columns correspond to time. Dashed vertical lines mark cue onsets (every 0.5 s from 0.5–3.0 s), the solid line marks delay start (3.5 s), and the white dotted line marks the start of the response period (4.0 s).

stimulus onsets, delay start, and response onset.

The resulting temporal decoding profiles reveal distinct roles for private and shared latent subspaces across the trial (Fig. 4.18). Across subspaces, decoding accuracy from shared latents increases progressively as the response period approaches, indicating that cross-area information integrates toward decision formation and movement execution. The shared subspace across all three regions shows a clear upward trend toward the response period, reaching its highest accuracy during the response phase and exhibiting modest increases following individual stimulus presentations. In contrast, the superficial SC private subspace achieves the highest prediction accuracy prior to the delay period, consistent with sustained maintenance of task-relevant signals during sensory processing. ALM private latents and the Deep SC–ALM shared subspace reach their peak decoding performance after the response, suggesting strong encoding of target side and decision variables, which are highly correlated in expert animals. The shared subspace between superficial and deep SC exhibits pronounced increases in prediction accuracy immediately following most stimulus onsets, pointing to early multisensory interactions between superficial and deep SC layers and a shared representation of stimulus direction. Collectively,

these temporal patterns reveal complementary functional roles across latent subspaces: private representations emphasize delay-period maintenance and response-related signals, while shared subspaces carry stimulus- and response-proximal information that increases as the behavioral response approaches.

4.7.4 Latent structure across task variables

In Fig. 4.20 and Fig. 4.21, we present all latents for the SC–ALM dataset, where we trial-average the latents within target side and stimulus type, respectively. In general, most latents exhibit oscillatory patterns that, to varying degrees, are tuned to stimulus presentation times; superficial SC latents tend to exhibit higher-frequency oscillations. Shared latents for Deep SC and ALM, and region-specific ALM latents, exhibit separation between the target sides during the delay and response. Superficial SC latents and shared superficial–deep SC latents exhibit separate activity patterns for tactile trials compared to visual and multisensory trials. In contrast, Deep SC and ALM shared latents exhibit patterns that separate all three stimulus types.

We next evaluate temporal decoding of the target variable (target side) from the latents to better determine when task information is represented in shared versus private subspaces.

4.7.5 Baseline comparisons

Comparison to Concatenated-Data Baseline

As a baseline, we used the same transformer autoencoder backbone to analyze concatenated data across all three regions (superficial SC, deep SC, and ALM), without enforcing a decomposition into shared and region-private subspaces. As in the M1–PMd concatenated baseline (Sec. 4.6.5), this model is trained using only the reconstruction loss (Eq. 4.8, applied across regions) and the orthogonality regularization (Eq. 4.11). The extracted latents were evaluated on discrete modality classification and continuous time decoding of target side.

The concatenated model can capture dominant task information for decoding, but the latents obtained in this baseline do not identify computations that are region-specific versus

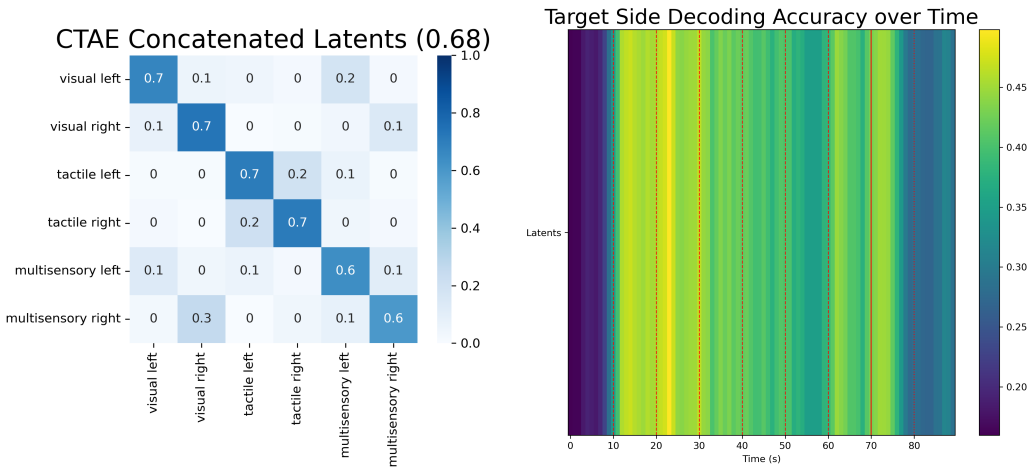


Figure 4.19. Concatenated-data decoding performance on the SC–ALM dataset. (a) Confusion matrix for single-trial stimulus-modality classification. True and predicted modalities are shown on the Y- and X-axes, respectively. (b) Continuous time decoding of target side from concatenated latents.

shared across regions. This is evident in Fig. 4.19, where the time-resolved decoding traces lack the structured subspace-specific patterns observed for CTAE (Fig. 4.18). Thus, while the concatenated model reflects task-relevant variance, it provides limited interpretability into the organization of distributed computations across the circuit.

Comparison to DeepCCA

We also compared against DeepCCA as a correlation-maximization baseline that learns a shared representation between pairs of regions but does not explicitly model temporal dynamics. Since DeepCCA is inherently pairwise, we applied it to the three region pairs: SupSC–DeepSC, DeepSC–ALM, and SupSC–ALM, matching each branch input dimension to the number of neurons in the corresponding region. We evaluated DeepCCA shared latents using the same downstream decoding analyses used throughout this chapter (Sec. 4.7.3

These results highlight that DeepCCA yields substantially weaker shared representations for behavioral decoding in this multisensory task. In addition, because DeepCCA does not model temporal structure, it provides limited interpretability for time-resolved analyses. In contrast, CTAE supports structured disentanglement into shared and region-private components while

Table 4.2. Pairwise DeepCCA results on SC–ALM compared to the corresponding CTAE pairwise shared subspaces. Entries show decoding accuracy for the stimulus × target-side classification task.

Region pair	DeepCCA	CTAE
SupSC–DeepSC	0.22	0.59
DeepSC–ALM	0.22	0.59
SupSC–ALM	0.20	0.39

capturing temporally organized latent dynamics that align with stimulus timing and decision formation (Figs. 4.15–4.18).

4.8 Training and Hyperparameter Details

To ensure consistency in model capacity across regions, we adopt identical causal Transformer-based encoder–decoder architectures for each region, using an equal number of encoder and decoder layers to maintain architectural symmetry. We then performed a grid search over:

Number of layers. We varied the total number of Transformer layers $L \in \{1,2,3\}$ in both encoder and decoder to trade off model capacity against overfitting risk and computational cost.

Latent dimensions. Private and shared dimensions, $d \in \{5,10,15\}$. Note - Here, the specified latent dimensionality d should be treated as an upper bound on bottleneck capacity. After training with the designed regularizers, redundant factors collapse to near-zero explained variance, yielding an effective dimensionality $d_{\text{eff}} \leq d$ that approximates the data's intrinsic dimensionality.

Loss weights.

- Shared-space: $\lambda_{\text{shared}} \in \{0,1,2\}$
- Alignment: $\lambda_{\text{align}} \in \{0,0.05,0.1,0.5\}$
- Orthogonality: $\lambda_{\text{orth}} \in \{0,0.001,0.01,0.05\}$

Learning rate. $\eta \in \{10^{-3},10^{-4}\}$

Warm-up schedule for orthogonality loss. To prevent an overly strict orthogonality constraint from hindering early representation learning, we set $\lambda_{\text{orth}} = 0$ for the first e epochs—allowing the model to focus on reconstruction and shared/private separation—then linearly ramp it up to its target λ_{orth} value over the next e epochs. This gradual increase stabilizes training by delaying the full orthogonality penalty until the autoencoders have already learned meaningful features. We explored $e \in \{100, 500, 1000\}$. Formally,

$$\lambda_{\text{orth}}^{(t)} = \begin{cases} 0, & t \leq e, \\ \dfrac{t-e}{e}\,\lambda_{\text{orth}}, & e < t \leq 2e, \\ \lambda_{\text{orth}}, & t > 2e\,. \end{cases}$$

Positional encoding. We use fixed, deterministic sinusoidal signals into each input embedding to convey temporal order, following [Vaswani et al., 2017]. Specifically, for time step t and embedding dimension index i (with model dimension d_{model}):

$$\text{PE}(t, 2i) = \sin\Big(\tfrac{t}{10000^{2i/d_{\text{model}}}}\Big), \quad \text{PE}(t, 2i+1) = \cos\Big(\tfrac{t}{10000^{2i/d_{\text{model}}}}\Big).$$

These encodings provide both absolute and relative position information without adding any learnable parameters, enabling the self-attention layers to distinguish different time steps.

All models were trained for $10,000$ epochs, and the run minimizing the total loss in (4.12) on a held-out validation set was selected. Table 4.3 summarizes all the hyperparameters and their corresponding search ranges and Table 4.4 reports the optimal settings for the two real and one synthetic datasets studied. For the corresponding DLAG and mDLAG baselines, we matched the total number of latent dimensions to those used by CTAE for each dataset (Table 4.4) to ensure a fair comparison.

Table 4.3. Hyperparameter search ranges.

Hyperparameter	**Values**
# Transformer layers L	$\{1,2,3\}$
latent dims. $d_1 = d_2 = d_s$	$\{5,10,15\}$
λ_{shared}	$\{0,1,2,5\}$
λ_{align}	$\{0,0.05,0.1,0.5\}$
λ_{orth}	$\{0,0.001,0.01,0.05\}$
Learning rate η	$\{10^{-3},10^{-4}\}$
Warm-up epochs for orth. loss	{100,500,1000}

Table 4.4. Selected hyperparameters for M1–PMd, SC-ALM and Synthetic datasets.

Parameter	**M1–PMd**	**SC–ALM**	**Synthetic**
latent dims	10	15	$d_s = 4, d_1 = d_2 = 3$
# Layers L	3	1	1
λ_{shared}	1	1	5
λ_{align}	0.5	0.1	0.1
λ_{orth}	0.01	0.01	0.01
Learning rate η	10^{-4}	10^{-4}	10^{-3}
Warm-up (orth. loss)	100 epochs	1000 epochs	100 epochs

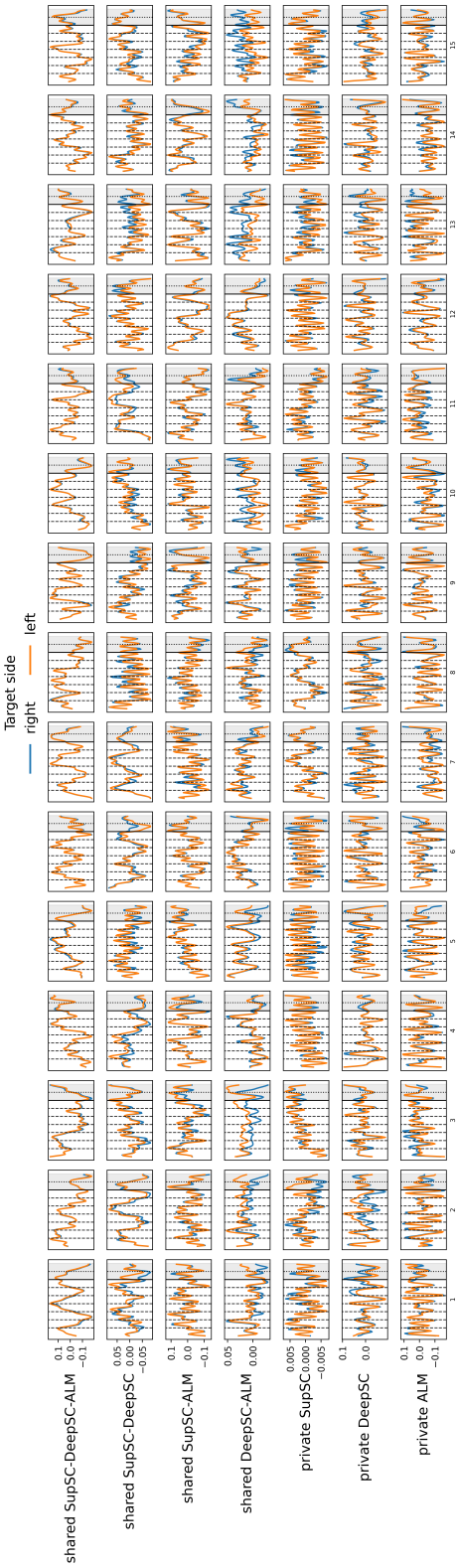


Figure 4.20. SC–ALM dataset. All CTAE latents averaged across target side.

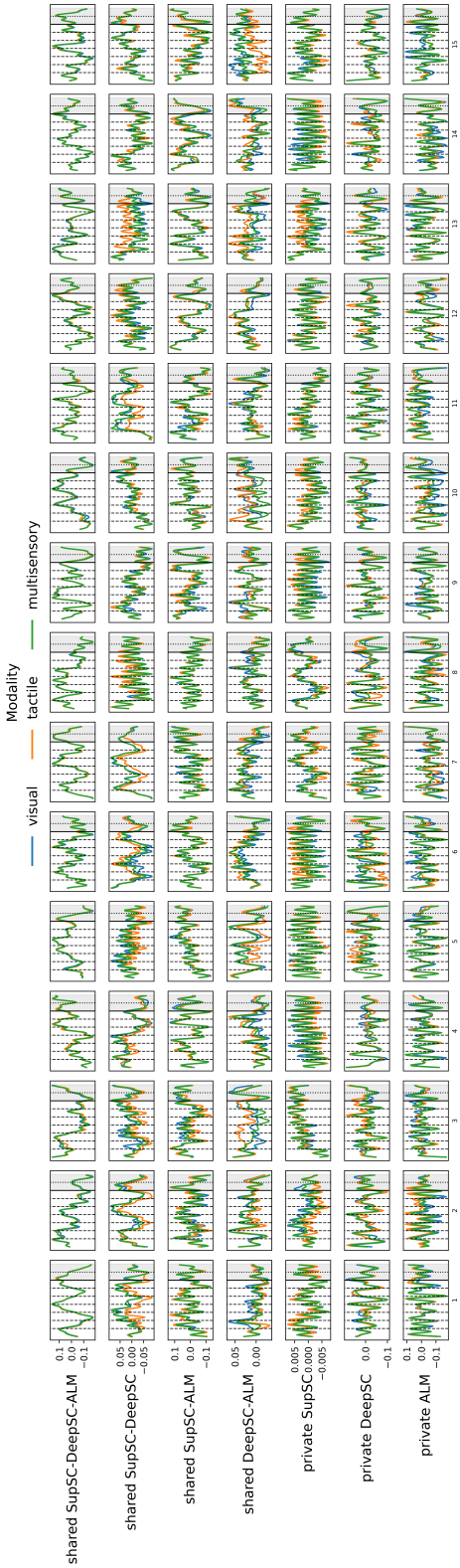


Figure 4.21. SC–ALM dataset. All CTAE latents averaged across stimulus type (visual, tactile, multisensory).

Acknowledgements

Chapter 4, in full, is a reprint of the following material: Sristi, Ram Dyuthi; Narasimha, Sowmya Manojna; Huang, Jingya; Despatin, Alice; Musall, Simon; Gilja, Vikash; Mishne, Gal (2026). "Coupled Transformer Autoencoder for Disentangling Multi-Region Neural Latent Dynamics." In *Proceedings of the International Conference on Learning Representations (ICLR)*. The dissertation author is the primary author of this paper.

Chapter 5

Complementary Cortical and Thalamic Contributions to Cell-Type-Specific Striatal Dynamics During Movement

5.1 Introduction

Generating appropriate motor outputs to effectively interact with the outside world is a major goal of the central nervous system. Recent advances in large-scale recordings have highlighted that many interconnected areas distributed throughout the brain have movement-related activity [Steinmetz et al., 2019, Allen et al., 2019, Chen et al., 2024, Khilkevich et al., 2024, Angelaki et al., 2025]. Yet the principles that govern information flow among brain areas, such as the function and specificity of long-range connections, are only beginning to be understood [Knowland et al., 2017, Economo et al., 2018, Hwang et al., 2019, Lee et al., 2020, Tervo et al., 2021, Yang et al., 2022c, Inácio et al., 2025]. Individual brain areas are themselves composed of diverse cell types, defined by their transcriptional and functional properties as well as anatomical connectivity [Saunders et al., 2018, Phillips et al., 2019, manuscript editors et al., 2021, Yao et al., 2023, Siletti et al., 2023]. Behavior-relevant information encoded in a given presynaptic area may be passed down uniformly across cell types or separately onto specific neuronal subpopulations of their postsynaptic targets. Characterization of neural circuits at this level of specificity is important for a deeper understanding of how coordinated activity among

motor brain areas is achieved and how it relates precisely to motor actions.

Within the cortico-basal-ganglia-thalamic loop involved in movement generation and execution [Graybiel, 1998, Redgrave et al., 1999, Yin and Knowlton, 2006, Dudman and Krakauer, 2016], the dorsolateral striatum (DLS) integrates excitatory inputs from both cortex and thalamus. Striatal direct- and indirect-pathway medium spiny neurons (dMSNs and iMSNs) send distinct axonal projections to the basal ganglia output nuclei via the direct or indirect pathways and have been associated with different roles during movement execution. It is believed that dMSNs play a role in the initiation and facilitation of selected motor sequences, whereas iMSNs are involved in switching between different sequences or inhibiting unwanted motor programs [Kravitz et al., 2010, Tecuapetla et al., 2016, Yttri and Dudman, 2016, Geddes et al., 2018, Sheng et al., 2019]. Previous studies have shown that both dMSNs and iMSNs are active during movement, but they exhibit distinct activity patterns [Cui et al., 2013, Jin et al., 2014, Meng et al., 2018, Parker et al., 2018, Markowitz et al., 2018, Cruz et al., 2022]. Whether these differences are inherited from their upstream inputs or shaped locally within the DLS is unknown.

In particular, among the cortical regions that project to the dorsolateral striatum [Wall et al., 2013, Hintiryan et al., 2016, Hunnicutt et al., 2016], primary and secondary (M1 and M2) motor cortices are critically involved in motor preparation, execution, and learning [Costa et al., 2004, Peters et al., 2014, 2017, Barthas and Kwan, 2017, Makino et al., 2017, Omlor et al., 2019]. Among the thalamic nuclei, the parafascicular nucleus of the thalamus (PF) represents the major input to both direct and indirect pathways [Wall et al., 2013, Hunnicutt et al., 2016, Mandelbaum et al., 2019] and participates in movement execution [Díaz-Hernández et al., 2018, Fallon et al., 2023, Sibener et al., 2025]. It is unknown how cortical and thalamic neurons contribute to the cell-type-specific activity of dMSNs and iMSNs, especially given that the degree of overlap in their connections with dMSNs and iMSNs is debated [Nelson et al., 2021, Klug et al., 2025].

We investigated the activity dynamics of dMSNs and iMSNs, as well as their cortical and thalamic inputs, by combining monosynaptically restricted trans-synaptic rabies virus with conditional gene expression together with 2-photon imaging, as mice performed a motor task. We

developed and applied a recurrent neural network (RNN) classifier on their activity to determine the degree of functional differences between the two striatal cell types and between their cortical and thalamic inputs. We identified heterogeneous functional subpopulations within MSNs and within their inputs, aligned with specific movement phases. Many dMSNs showed activity around movement initiation and termination, while more iMSNs were active during the movement. The activity of their cortical and thalamic inputs exhibited contrasting dynamics, suggesting a complementary contribution of these inputs to striatal activity. However, within each input region, dMSN-projecting and iMSN-projecting populations showed similar activity patterns. Notably, activity related to forelimb positions was predominantly found in thalamocortical neurons, especially those projecting to dMSNs, suggesting that kinematic information might be conveyed to MSNs in a cell-type-specific manner.

5.2 Experimental Design and Data Acquisition

5.2.1 Animal Models and Genetic Targeting

All animal procedures were performed in accordance with guidelines and protocols approved by the UCSD Institutional Animal Care and Use Committee and the National Institutes of Health (NIH). Mice were group-housed in disposable cages with standard bedding in a temperature-controlled room (21°C) with a reversed light cycle (10.00–22.00 corresponded to the dark period). Mice were allowed ad libitum access to food and water. Experiments were performed in two transgenic mouse-lines: B6.FVB(Cg)-Tg(Adora2a-Cre)KG139Gsat/Mmucd (RRID:MMRRC_036158-UCD) and B6.FVB(Cg)-Tg(Drd1-Cre)FK150Gsat/Mmucd (RRID: MMRRC_036916-UCD). Both males and females were used for experiments. All experiments were performed during the dark period.

5.2.2 Surgical Procedures and Viral Injections

For all surgeries, adult mice (6-10 weeks old) were anesthetized with 1-2% isoflurane and injected with Baytril (10 mg/kg), dexamethasone (2 mg/kg), and buprenorphine (0.1 mg/kg) subcutaneously at the beginning to prevent infection, inflammation, and pain. Recovery was monitored daily for 3 consecutive days following surgery with administration of Baytril, dexamethasone, and buprenorphine.

For 2-photon imaging of dMSNs and iMSNs in the dorsolateral striatum, AAV1-hSyn-FLEX-GCaMP6f (Addgene, Catalog # 100833-AAV1) was injected via beveled glass pipettes into the right hemisphere of Drd1-Cre and Adora2a-Cre mice respectively, at coordinates: 0.5 mm anterior and 2.2 mm lateral from bregma, at depths 2.2, 2.3 and 2.4 mm from the pia (200 nL at each depth, at a rate of 20 nL per minute. The pipette was kept in the brain for 15 minutes to avoid virus leakage. For GRIN lens implantation in DLS (or PF, see below), a similar procedure to [Resendez et al., 2016] was followed. GRIN lenses of 0.5 or 0.6 mm diameter (Inscopix, product number 1050-006232 and 1050-004597) were lowered into the above-mentioned coordinates at a depth of 2 mm for DLS imaging. GRIN lens implantation was performed within a week from viral injection. Imaging was usually performed 3-4 weeks after lens implantation.

For 2-photon imaging of the cortical or thalamic inputs that project onto dMSNs or iMSNs in the dorsolateral striatum, a 1:1 or 1:2 solution mix of helper viruses AAV-EF1a-DIO-mRuby2-P2A-TVA and AAV-EF1a-DIO-oPBG was injected into the right hemisphere of Drd1-Cre and Adora2a-Cre mice respectively, at coordinates: 0.5 mm anterior and 2.2 mm lateral from bregma, at depths 2.2, 2.3 and 2.4 mm from the pia (0.2 µL at each depth) at each depth). In the case of M1 imaging, a 22° angled injection into the DLS was performed in order to avoid inserting the pipette directly through M1, at coordinates: 0.5 mm anterior, 3.17 mm lateral from bregma. On the same day, a craniotomy centered around 0.5 mm anterior and 1.5 mm lateral from bregma or around 2.3 mm anterior and 1.2 mm lateral from bregma, was performed

as previously described [Peters et al., 2014, Makino et al., 2017, Komiyama et al., 2010, Ren et al., 2022] for imaging M1 or M2, respectively. Otherwise for PF imaging, a GRIN lens was lowered into the brain at coordinates: 2.2 mm posterior and 0.65 mm lateral from bregma at a depth of 3.2 mm. After 3-4 weeks, EnvA-pseudotyped rabies that expresses GCaMP6f (EnvA-RVdG-mRuby-GCaMP6f, in the case of M2 imaging or EnvA-RVdG-GCaMP6f in the case of M1 and PF imaging) was injected into DLS at the same coordinates and depths as mentioned above 0.2 µL at each depth. At a 22° angle for M1 imaging). After virus injection, the glass pipette was left in place for 20-30 minutes to minimize off-target expression. Imaging with rabies virus was limited to 5-9 days from infection (49). AAV helper and EnvA-pseudotyped rabies viruses were produced in Prof. Lim's laboratory at UCSD.

5.2.3 Motorized ladder task

Head-fixed mice were positioned on a custom-built, circular ladder (diameter = 18 cm, rung diameter = 0.3 cm, rung spacing = 1 cm) motorized by an electric DC motor (12V, 60 RPM). Trial structure was controlled using BPod v0.5. In each trial, 8 seconds of ladder rotation (speed 1 0 mm/s) followed an auditory cue (1 s at 3 kHz). Trials were interleaved by a variable 6-8 second inter-trial interval. In each trial, the auditory cue (= trial start) followed a 1 00 ms infrared LED light that was placed outside of the mouse's view and used for aligning trials during video analysis. Mice were trained on the ladder task for 7 to 12 days, daily, till they became able to walk or run on the rungs by engaging all four limbs. For the experiments where Enva-pseudotyped rabies was used for labelling MSN inputs, mice were trained on the task for 4-5 days before the viral injection, and continued to be trained after, and imaged within 9 days from the viral injection. During 2-photon imaging sessions, two orthogonal views of the mouse were captured simultaneously through a mirror mounted at 45° inside the motor task setup, at 30 Hz with an IR-sensitive video camera (DMK 23U618, The Imaging Source) controlled by IC Capture software, version 2.5.

5.2.4 In Vivo Calcium Imaging

Imaging data acquisition

2-photon imaging was conducted with a commercial two-photon microscope (Movable objective microscope (MOM), Sutter Instrument, retrofitted with a resonant galvanometer-based scanning system from Thorlabs), 16× objective (Nikon) and a Ti:Sapphire excitation laser at 925 nm excitation light (Mai Tai, Spectra-Physics) controlled by ScanImage (Vidrio Technologies). For imaging of M1, DLS and PF neurons, images (512×512 pixels, 0.98 μm/pixel) were recorded at 28 Hz for the duration of the behavioral session (20 minutes). For M2 imaging, two fields of view at different depths were acquired near-simultaneously at 14 Hz. Frame times were recorded and synchronized with behavioral recordings by the Ephus software (controlled by MATLAB). Multiple non-overlapping fields of view were recorded from each animal in consecutive days.

Two-photon imaging analysis

Images were first aligned frame-by-frame using a custom MATLAB program to correct for lateral shifts [Mitani and Komiyama, 2018]. ROIs were manually drawn using a custom MATLAB program [Peters et al., 2014, Ren et al., 2022]. A ring-shaped background-ROI containing neuropil signal was created from the border of each neuronal ROI to a width of 6 pixels. Fluorescence was processed as described [Peters et al., 2014]. Briefly, pixels within each ROI were averaged to create a fluorescence time series, and the background fluorescence was subtracted. The time-varying baseline (F) of the fluorescence trace was estimated by iteratively smoothing inactive portions of the trace. The normalized dF/F trace was then calculated, where dF was determined by subtracting the baseline trace from the raw trace and F is the calculated time-varying baseline. In the case of M2 imaging, dF/F was calculated as the ratio between the GCaMP6f and the mRuby signal. Two criteria were applied for including the recorded neurons in all subsequent analyses: 1) session median raw fluorescence signal above 100 a.u. and 2) at least two activity events (see below) per minute present during the imaging session.

5.2.5 Muscimol inactivation

Muscimol inactivation experiments were performed on a subset of mice implanted with a GRIN lens in DLS. A field of view in DLS was imaged for approximately 15 minutes as mice performed the ladder task. Immediately after the imaging session, mice were anesthetized as described above and 25 nL muscimol hydrobromide (5 $\mu g\,\mu L^{-1}$, Sigma-Aldrich) was injected with a glass pipette unilaterally into the right hemisphere, at these coordinates: M1: 0.5 mm anterior and 1.5 mm lateral from bregma, depth 0.35 mm; M2, 2.3 mm anterior and 1.2 mm lateral from bregma, depth 0.35 mm; PF: 2.2 mm posterior and 0.65 mm lateral from bregma, depth of 3.3 mm. Mice were allowed to recover in their home cage and the same field of view was re-imaged after 30 min from the injection.

5.2.6 Activity events

For quantifying the effect of muscimol inactivation of inputs on striatal activity, activity events were determined as previously described [Peters et al., 2014] and compared with the imaging sessions before the manipulation. Briefly, events detected sharp rises in the fluorescence trace and were defined if the first derivative (velocity) of the fluorescence trace crossed five times the standard deviation of the inactive portions of the fluorescence trace. Trials that contained at least one activity event were defined as 'active trials', and the fraction of active trials was compared before and after the manipulation experiment.

5.2.7 Histology and Anatomical Verification

At the end of the imaging sessions, mice were transcardially perfused with sodium phosphate buffered saline (PBS) followed by 4% paraformaldehyde (PFA, Sigma-Aldrich) in PBS. Brains were extracted, postfixed in 4% PFA overnight at 4°C and then stored in 30% sucrose in PBS at 4°C for 2-3 days. Coronal sections (40 µm thick) were obtained using a Leica SM 2000R sliding microtome. Sections were mounted on slides using CC Mount (Sigma-Aldrich) and imaged with an Axiozoom V16 microscope to confirm viral expression and GRIN lens

positioning.

5.3 Computational Methods

Notations: We use the following notational convention in this chapter. Upper case non-bold for fixed scalars N, Upper case bold for matrices $\mathbf{M}$, and lower case bold for vectors $\mathbf{v}$.

5.3.1 Trial Ensemble Attention Network (TEA-net)

Data preprocessing and trial-ensemble construction

For every neuron, dF/F traces were segmented into single-trial windows that started 2 sec before the ladder onset and ended 4 sec after ladder offset. For each region we determined N, the smallest number of trials recorded in any session for a given region (DLS: $N = 42$, M1: $N = 47$, M2: $N = 50$, PF: $N = 28$). For each neuron we then generated multiple bootstrap ensembles. We randomly sampled N trials without replacement and stacked them into a matrix $\mathbf{X} \in \mathbb{R}^{N \times T}$ to form an ensemble of the neuron's activity of T timeframes across N trials. The number of ensembles used to train TEA-Net was balanced across cell types in each region to ensure equal class priors. In a given region, we define N_{all} as the total number of neurons, with N_d dMSNs (or dMSN-projecting) and N_i iMSNs (or iMSN-projecting) where $N_d + N_i = N_{\text{all}}$. We generated $N_{\text{all}} \times 200$ ensembles in total, allocating half ($N_{\text{all}} \times 100$) to each cell type. Thus, every dMSN (or dMSN projecting) neuron contributed $\left(\frac{N_{\text{all}} \times 100}{N_d}\right)$ ensembles and every iMSN (or iMSN projecting) neuron contributed $\left(\frac{N_{\text{all}} \times 100}{N_i}\right)$ ensembles, ensuring balanced training data for TEA-net.

Network architecture

Given an input ensemble $\mathbf{X}$, TEA-net predicts whether its source is a direct or indirect pathway neuron. The network consists of three modules:

1. *Temporal encoder* – a single-layer vanilla recurrent neural network (hidden size $= d_h = 10$) processes the ensemble column-wise, i.e. the input x_t at every timeframe t consists of the

activity of the neuron at time t across the N trials. The encoder outputs a hidden state $\in \mathbb{R}^{d_h \times T}$, whose t-th column is the vector $\mathbf{h}_t$.

2. *Learned temporal attention* – We use a lightweight attention approach, analogous to (50), by applying a 1×1 convolution across hidden channels to produce a scalar relevance score e_t for every frame. Soft-max normalization yields attention weights α_t that highlight informative hidden representations while down-weighting uninformative ones.

3. *Context aggregation and classification* – the weighted hidden states $(\alpha_t \cdot \mathbf{h}_t)$ are summed over the hidden dimension, producing a length-T context vector $\mathbf{h}'_t = \mathbf{1}^T(\alpha_t \mathbf{h}_t)$ that preserves the temporal nature of the ensemble. A fully connected layer (input $= T$, output $= 2$) maps $\mathbf{h}'_t$ to class logits.

Training procedure

Training and evaluation followed a 10-group fold cross-validation at the session level: all neurons recorded in the same imaging session were assigned to the same fold, and every fold contained $\sim 10\%$ of the direct- and indirect-pathway neurons. Models were trained for 50 epochs with a batch size of 256 ensembles, an Adam optimizer [Adam et al., 2014] (initial learning rate $= 0.001$, weight decay $= 0.001$), and a step-decay scheduler that reduced the learning rate by 90% every 10 epochs. The loss function was cross-entropy. Early stopping [Prechelt, 2012] (patience $= 5$ epochs) was applied based on validation loss to prevent over-fitting.

Evaluation Metrics

For each test neuron in each cross fold, the majority vote across its 200 ensemble predictions determined the final label. We then compute the average cell type accuracy of the classifier for each region, e.g., for DLS this is given by:

$$0.5 \times \left(\frac{\#\text{correctly classified } d\text{MSNs}}{\#\text{ total } d\text{MSNs}} + \frac{\#\text{ correctly classified } i\text{MSNs}}{\#\text{ total } i\text{MSNs}} \right)$$

To test whether classifier performance exceeded chance, cell-type labels were randomly reassigned within each session, the entire cross-validation procedure was repeated, and the resulting accuracies were compared to the true-label distribution with a two-sided Mann–Whitney U-test.

Other classifier models:

We trained two baseline models and evaluated them with the same metrics used for TEA-Net, enabling a direct comparison.

Support-vector-machine (SVM) baselines

As a linear reference, we trained *linear SVMs* [Cortes and Vapnik, 1995] on two alternative representations of the data:

- Trial-averaged activity: the dF/F trace averaged across all the trials of a neuron, yielding a single T-length vector per neuron. The soft-margin penalty hyper-parameter C was tuned by 10-fold cross-validation, searching the grid $C \in \{0.1, 1, 10, 10^2, 10^3, 10^4, 10^5\}$; the best score was obtained at $C = 10$.

- Trial-ensemble input: the random ensemble matrix $\mathbf{X}$ flattened into a vector. The same grid search favored a larger penalization constant, $C = 10^4$.

Feed-forward control network with attention (FC-attention)

To test whether TEA-net's performance benefits from its ability to model temporal dependencies, we built a parameter-matched baseline in which the recurrent temporal encoder is replaced by a time-distributed fully connected linear layer. Concretely, each frame $\mathbf{x}_t \in \mathbb{R}^N$ is mapped to a hidden vector $\mathbf{h}_t = \mathrm{ReLU}(\mathbf{W}\mathbf{x}_t + \mathbf{b})$ where $\mathbf{W} \in \mathbb{R}^{d_h \times N}$, with the same weights applied independently at every t. All subsequent stages—learned temporal attention, context aggregation, and classification—are identical to those in TEA-net and are applied to the sequence $\{\mathbf{h}_t\}$. Thus, the performance gap quantifies the benefit of the recurrent model (i.e. it is not due to additional capacity or alternative readout mechanisms).

5.3.2 Hierachical clustering of latent space trajectories

To uncover functional sub-groups of MSNs and their inputs, we clustered the latent space representation learned by TEA-net for each region. To derive a robust, fold-independent embedding for every neuron, we first averaged over trial ensembles within each cross-validation fold: TEA-net projects every bootstrap trial ensemble to a T-dimensional latent trajectory, and the element-wise mean of these trajectories yields a single T-length vector that summarises the fold. Repeating this procedure for all ten folds produces ten such vectors, each residing in the latent space of its respective model. After min–max normalizing each vector within its own fold, we concatenated the ten vectors, resulting in a $10 \times T$ matrix that serves as the neuron's final latent representation for subsequent clustering. Pairwise dissimilarities between these vectors were computed with Euclidean distance, and agglomerative hierarchical clustering was performed with Ward's criterion, which merges clusters to minimize the increase in within-cluster [Ward Jr, 1963]. Applying Dynamic Tree Cut—specifically the `cutreeDynamic` function from the R package `dynamicTreeCut` [Langfelder et al., 2008]—to the resulting dendrogram yielded the cluster assignments.

Hyper-parameter selection (*deepSplit* and *minClusterSize*)

The two parameters governing dynamic tree cutting are 1) *deepSplit* which governs the granularity of cluster division and 2) *minClusterSize* which specifies the size of the smallest allowable cluster as a fraction of the total neuron count. These two parameters were tuned with a 10000-iteration sub-sampling stability analysis. For each brain region, we repeatedly clustered bootstrap subsets containing 80 % of the neurons. We then selected the parameter pair that maximized the fraction of subsets whose cluster count matched the full-data solution, i.e. the mode of the sub-sampling histogram coincided with the cluster number at 100 % sampling. The resulting region-specific settings were: for DLS, *deepSplit*=3 and *minClusterSize*=0.095; for M1, *deepSplit*=2 and *minClusterSize*=0.080; for M2, *deepSplit*=3 and *minClusterSize*=0.070; for PF, *deepSplit*=3 and *minClusterSize*=0.080.

Cluster accuracy and cell-type dominance

Two complementary metrics were computed for every cluster returned by the hierarchical procedure: a *normalized classification accuracy* that evaluates how well TEA-net's cell-type predictions align with ground truth within that cluster, and a *cell-type-dominance index* that quantifies the cluster's compositional bias toward dMSNs (or dMSN-projecting) or iMSNs (or iMSN-projecting) neurons.

We define the following cell type normalized fractions:

$$p_{d,cc} = \frac{\text{\#correctly classified dMSNs in cluster}}{\text{\#total dMSNs in region}}, \quad p_{i,cc} = \frac{\text{\#correctly classified iMSNs in cluster}}{\text{\#total iMSNs in region}}$$

$$p_{d,all} = \frac{\text{\#dMSNs in cluster}}{\text{\#total dMSNs in region}}, \quad p_{i,all} = \frac{\text{\#iMSNs in cluster}}{\text{\#total iMSNs in region}}$$

where the subscripts d and i represent dMSNs and iMSNs (dMSN-projecting and iMSN-projecting for the input regions), respectively, while 'cc' indicates correctly classified neurons.

The cluster's accuracy, adjusted for unbalanced cell types in a given region, was then defined as:

$$\text{Acc}_{\text{norm}} = \frac{p_{d,cc} + p_{i,cc}}{p_{d,all} + p_{i,all}}$$

Chance level was obtained by randomly permuting the predicted labels across all neurons (10000 iterations) while keeping cluster membership and total cell-type counts unchanged. One-sided p-values were derived from this null distribution, and clusters with $p < 0.05$ were deemed statistically significant.

Cell type dominance index was defined as:

$$D = \frac{p_{d,all}}{p_{d,all} + p_{i,all}}$$

Values above 0.5 indicate dMSN (or dMSN-projecting) dominance, whereas values below 0.5 indicate iMSN (or iMSN-projecting) dominance. Chance level was assessed by shuffling the true cell-type identities across neurons (10000 iterations), recomputing the index, and deriving two-sided p-values.

Clusterability analysis

To quantify how strongly the data deviates from uniform random distribution, we computed the Hopkins statistic H [Hopkins and Skellam, 1954] for 1) the raw neural activity z-scored after trial-averaging and 2) the normalized TEA-net latent representations used for hierarchical clustering. The test was run with `pvclustend` (v 1.3; function `pvclustend.hopkins`). For a dataset of n neurons, the routine samples $m = 0.1\,n$ observations—half uniformly random "probe" points drawn from the data hyper-cube and half actual data points. It then compares their nearest-neighbor Euclidean distances, and returns the Hopkins statistic H. Values of H are interpreted as follows: $H \approx 0.5$ indicates complete spatial randomness, $H > 0.5$ increasing clusterability, and $H < 0.5$ regular (over-dispersed) structure. For each brain region we generated 10000 bootstrap subsamples (size = 80% of neurons) and evaluated H for the two feature spaces independently.

Cluster-similarity analysis across brain regions

We compared the cluster-average activity profiles across brain regions with a pair-wise Pearson correlation analysis. For every cluster $c_k^{(r)}$ obtained in region r we first computed an average dF/F trajectory, $m_k^{(r)}(t)$ by (1) averaging each neuron's activity across trials and (2) averaging across all neurons within the cluster. Given two regions, r_1 and r_2, we calculated an n_{r_1} and n_{r_2} similarity matrix whose (k,l) entry was the Pearson correlation between $m_k^{(r_1)}(t)$ and $m_l^{(r_2)}(t)$, where n_r is the number of clusters in region r.

5.3.3 Forelimb and neuronal rhythmicity analysis

To determine whether MSNs and their cortical and thalamic inputs exhibit rhythmic activity that match the cadence of the contralateral (left) forelimb during locomotion, we followed a three-step procedure encompassing limb tracking, spectral decomposition, and peak-based classification.

Video pre-processing and limb trajectory extraction

Video frames were acquired at 30 Hz and analyzed offline with DeepLabCut (DLC) (52) to extract the *x* and *y* coordinates of the four limbs. Several DLC models were trained on hundreds of video frames from different animals and recording sessions, to account for variabilities due to animal posture and locomotion idiosyncrasies, light conditions, and limb visibility on the mirror. The four limbs reflected on the mirror were labeled. After automatic labeling, in videos with missing frames, limb coordinates were linearly interpolated based on nearby frames. Labelled values that fell more than four standard deviations outside of the X and Y distributions were likewise replaced with linearly interpolated values. For the rhythmicity analysis we focused on the left forelimb. Because the forelimb mainly oscillates along a single linear axis related to the direction of movement during locomotion, the two-dimensional trajectory was reduced to a one-dimensional displacement by principal component analysis (PCA): the first principal component (PC_1) was taken as the scalar limb-position time-series for all subsequent analyses, where the sign of the PC was set so that positive values corresponded to forward motion.

Power-spectral density (PSD) estimation

Spectral content of both forelimb trajectory and neural activity was quantified with the Welch's method (SciPy v1.12.0), using a Hann window of 1024 samples with 50% segment overlap, which yields a frequency resolution of 0.03 Hz. The limb-position trace and neuronal activity were z-scored at each trial (starting at ladder onset and ending 1 sec after the ladder offset). Trial-wise PSDs were averaged to obtain one spectrum per neuron or per forelimb

trajectory.

Identification of forelimb-related rhythmic activity

For each session, the averaged limb-position PSD was examined to identify the most prominent spectral peak above 0.03 Hz, yielding the limb's cadence frequency f_{beh}. The locomotion-related frequency band, defined as the peak's full width measured at 10% of its maximum amplitude, served as the reference band for determining forelimb-related neuronal rhythmicity. Within this band, each neuron's PSD was examined to identify local maxima whose prominence (the height of a peak measured relative to the higher of its two neighboring troughs) exceeded a region-specific threshold that we determined by visual inspection of representative spectra: DLS ≥ 0.80, PF ≥ 1.00, M1 ≥ 0.65 and M2 ≥ 0.65 (PSD units). A neuron was labelled *rhythmic* if its most prominent peak lay inside the band and satisfied the corresponding prominence criterion. Proportions of rhythmic cells were compared between dMSNs and iMSNs, and between projection-defined cortical/thalamic populations, using two-sided Fisher's exact tests. P-values were Bonferroni-corrected to account for multiple comparisons.

Phase-tuning analysis

Phase tuning was determined only for neurons deemed to have forelimb-related rhythmic activity identified as above.

Stride detection and phase definition.

The one-dimensional paw-position trace (PC_1; see "limb trajectory extraction") was scanned for successive local minima and maxima. A stride was defined as the interval from one backmost excursion (minimum) to the next, with the intervening frontmost excursion (maximum) marking mid-stride. This interval was mapped onto a phase domain such that the first minimum corresponded to 0° (stride onset), the maximum to 180°, and the subsequent minimum to 360° (stride completion). Phases 0–180° therefore constituted the swing period, when the paw travels forward through the air, whereas phases 180–360° constituted the stance period, when the paw

contacts the ladder and moves posteriorly as the body advances relative to the ladder.

Stride and neural activity warping.

Because individual strides vary in duration, each detected stride was resampled to 360 equally spaced phase points by linear interpolation. The simultaneously recorded neural activity was interpolated onto the same phase grid, yielding one phase-locked activity vector per stride.

Phase tuning curve.

Within each trial, all stride-aligned activity vectors were averaged point-wise to obtain a trial-specific phase-tuning curve, $T_{\text{trial}}(\theta)$, with θ ranging from 0° to 360°. The phase of maximal activity, $\theta_{\text{peak,trial}} = \arg\max_{\theta} T_{\text{trial}}(\theta)$, was taken as the neuron's preferred phase on that trial. Preferred phases from all trials of a neuron were combined with a circular mean to give the neuron's overall preferred phase, $\theta_{\text{pref}} = \text{circmean}(\theta_{\text{peak,trial}})$. We compared preferred-phase distributions between neuron groups using the Watson two-sample U^2 test, evaluating significance ($\alpha = 0.05$) with an exact p-value obtained from 10000 random permutations of the group labels for every comparison [Watson, 1962, Landler et al., 2021].

Statistics

Statistical tests were selected based on data distributions and significance was set at $p < 0.05$. Non-parametric tests were used throughout the manuscript where appropriate. No statistical methods were used to pre-determine sample sizes.

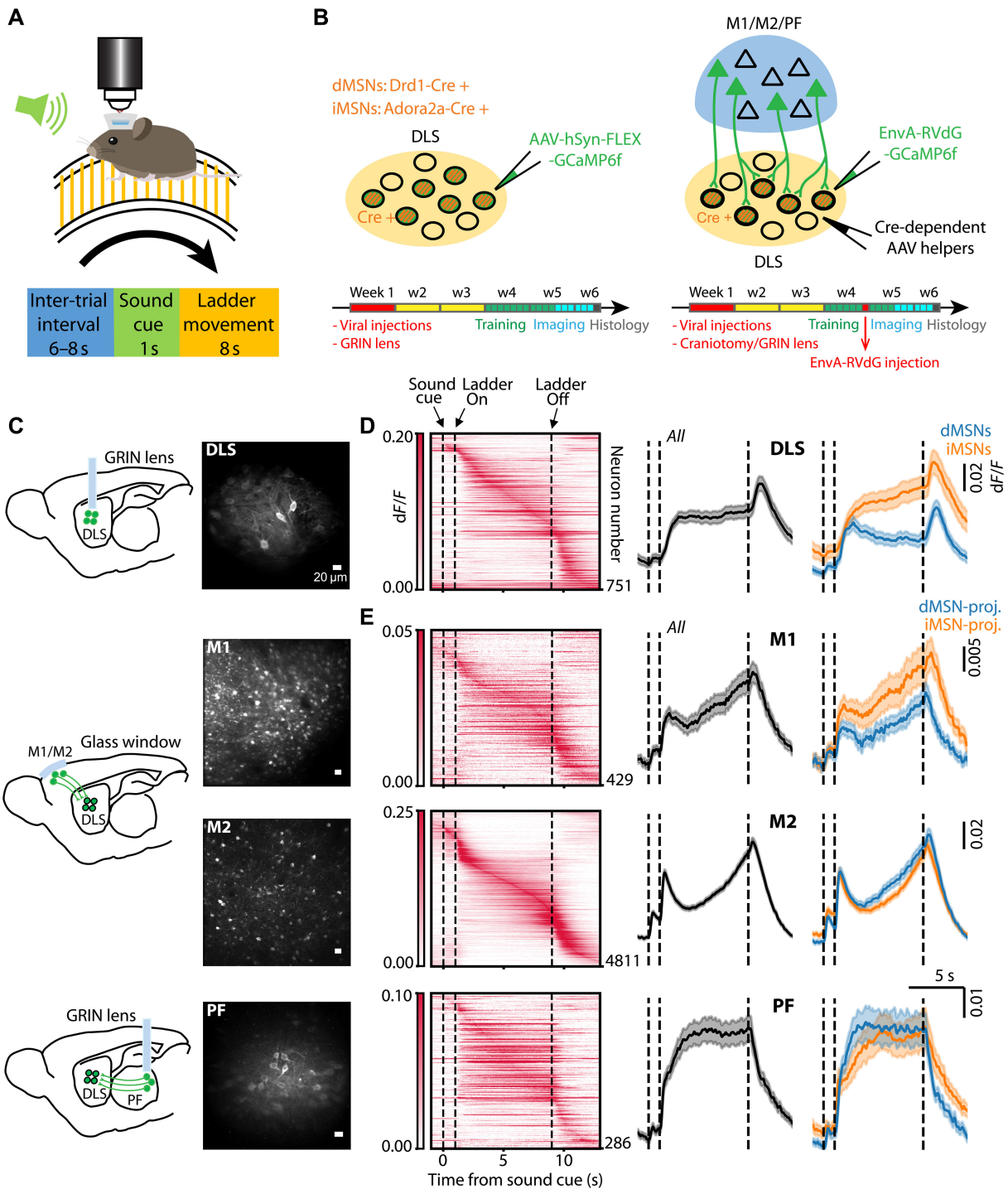
5.4 Results

5.4.1 Population activity patterns of dMSNs and iMSNs and their cortical and thalamic inputs

To characterize the activity patterns of dMSNs and iMSNs in the DLS, and of the neuronal subpopulations within the cortex and thalamus that project onto them, we conducted *in vivo* two-photon imaging in head-fixed mice while they walked on a motorized circular ladder. In each trial, an auditory cue preceded ladder rotation at a constant speed (Fig. 5.1A). To selectively label and

image the activity of dMSNs and iMSNs, a Cre-dependent viral vector of GCaMP6f was injected in the DLS of specific Cre lines (Drd1-Cre for dMSNs, Adora2a-Cre for iMSNs). To record the activity of cortical or thalamic neurons that project to dMSNs or iMSNs, Cre-dependent helper viruses that allow infection and trans-synaptic spread of monosynaptic pseudo-typed rabies virus (48) were injected in the DLS of D1R-Cre and A2A-Cre lines (Fig. 5.1B). After 3–4 weeks, EnvA-pseudotyped rabies that expresses GCaMP6f was injected in DLS (Additional Fig. 5.2). Imaging with rabies virus was limited to 5–9 days from infection [Wickersham et al., 2007a]. Cortical neurons were imaged through a chronic glass window, while striatal and thalamic neurons were imaged via a GRIN lens (Fig. 5.1C). Muscimol injection in each input region led to almost complete abolishment of the movement-related striatal activity (Additional Fig. 5.3, paired Wilcoxon rank-sum test, $p < 0.001$ for all comparisons after vs. before manipulation), confirming that these three inputs are required for driving DLS activity during the ladder task.

Figure 5.1. Population activity patterns of dMSNs and iMSNs and their cortical and thalamic inputs.
A) Schematic of the motor task. **B)** Left, schematic of the approach used for labeling dMSNs and iMSNs in DLS. Bottom, timeline of experiments. Right, schematic of the approach used for labeling the presynaptic cortical and thalamic inputs to dMSNs and iMSNs. **C)** Schematic of the imaged brain regions and maximum intensity projection images from 2-photon in vivo imaging, showing neurons that express GCaMP6f in DLS, M1, M2, and PF. Scale bar: 20μm. **D)** Left, trial-averaged activity heatmaps of neurons in DLS, ordered based on the maximum peak time of their activity. Middle, population average activity of all DLS neurons. Right, population average activity of dMSNs (blue) and iMSNs (orange), mean $\pm$ SEM. n = 337 dMSNs (from N = 21 mice) and n = 414 iMSNs (from N = 12 mice). **E)** Same as D) for the input neurons in M1, M2 and PF. M1: n = 180 dMSN-projecting neurons (from N = 5 mice) and 249 iMSN-projecting neurons (from N = 4 mice); M2: n = 1849 dMSN-projecting neurons (from N = 12 mice) and 2962 iMSN-projecting neurons (from N = 12 mice); PF: n = 136 dMSN-projecting neurons (from N = 8 mice) and 150 iMSN-projecting neurons (from N = 9 mice). DLS: dorsolateral striatum; M1: primary motor cortex; M2: secondary motor cortex; PF: parafascicular nucleus.


A
Inter-trial interval 6–8 s
Sound cue 1 s
Ladder movement 8 s
B
dMSNs: Drd1-Cre +
iMSNs: Adora2a-Cre +
DLS
AAV-hSyn-FLEX -GCaMP6f
Cre +
Week 1 w2 w3 w4 w5 w6
- Viral injections
- GRIN lens
Training Imaging Histology
M1/M2/PF
EnvA-RVdG -GCaMP6f
Cre-dependent AAV helpers
- Craniotomy/GRIN lens
EnvA-RVdG injection
C
GRIN lens
DLS
20 µm
M1/M2
Glass window
M1
M2
PF
D
Sound cue
Ladder On
Ladder Off
dF/F
Neuron number
All
dMSNs
iMSNs
0.02 dF/F
E
dMSN-proj.
iMSN-proj.
0.005
0.02
5 s
0.01
Time from sound cue (s)

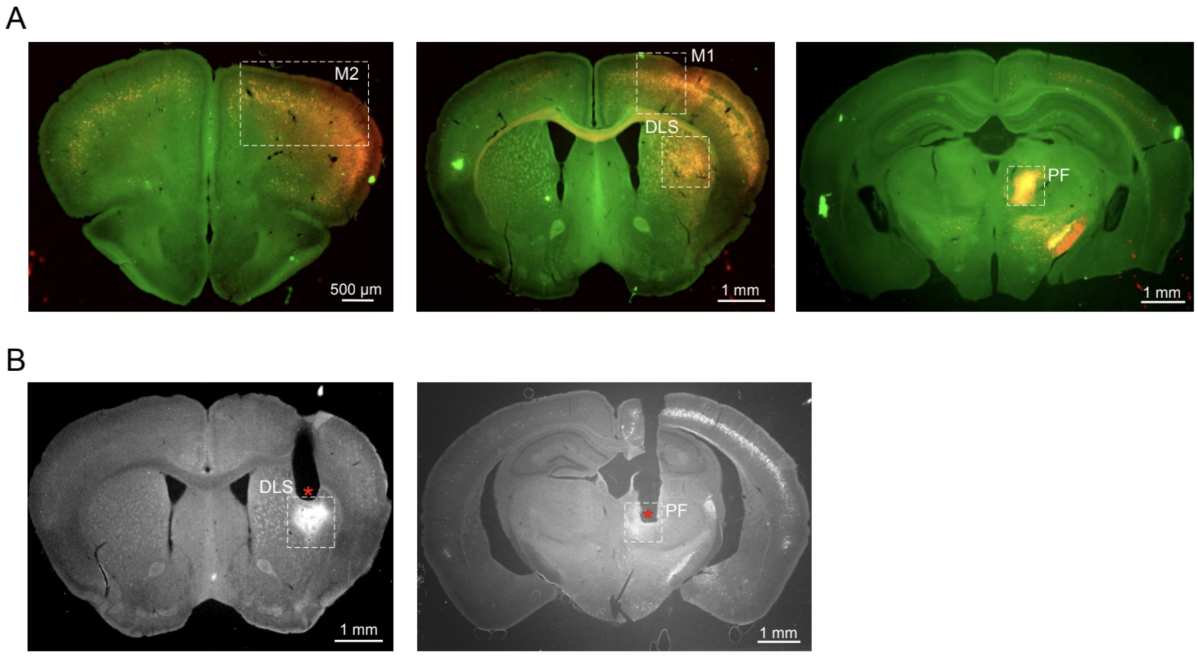


Figure 5.2. Example expression of mRuby-GCaMP6f and GRIN lens positioning.
A) Example coronal brain sections from a Drd1-Cre mouse that show the expression of mRuby-GCaMP6f (mRuby signal in red, GCaMP6f signal in green) via the EnvA-pseudotyped rabies in DLS (injection site) and in the three input regions imaged in this study: M1, M2 and PF. **B)** Example coronal brain sections that show the positioning of the GRIN lens in DLS (left) or in PF (right). Red asterisk indicates the GRIN lens bottom.

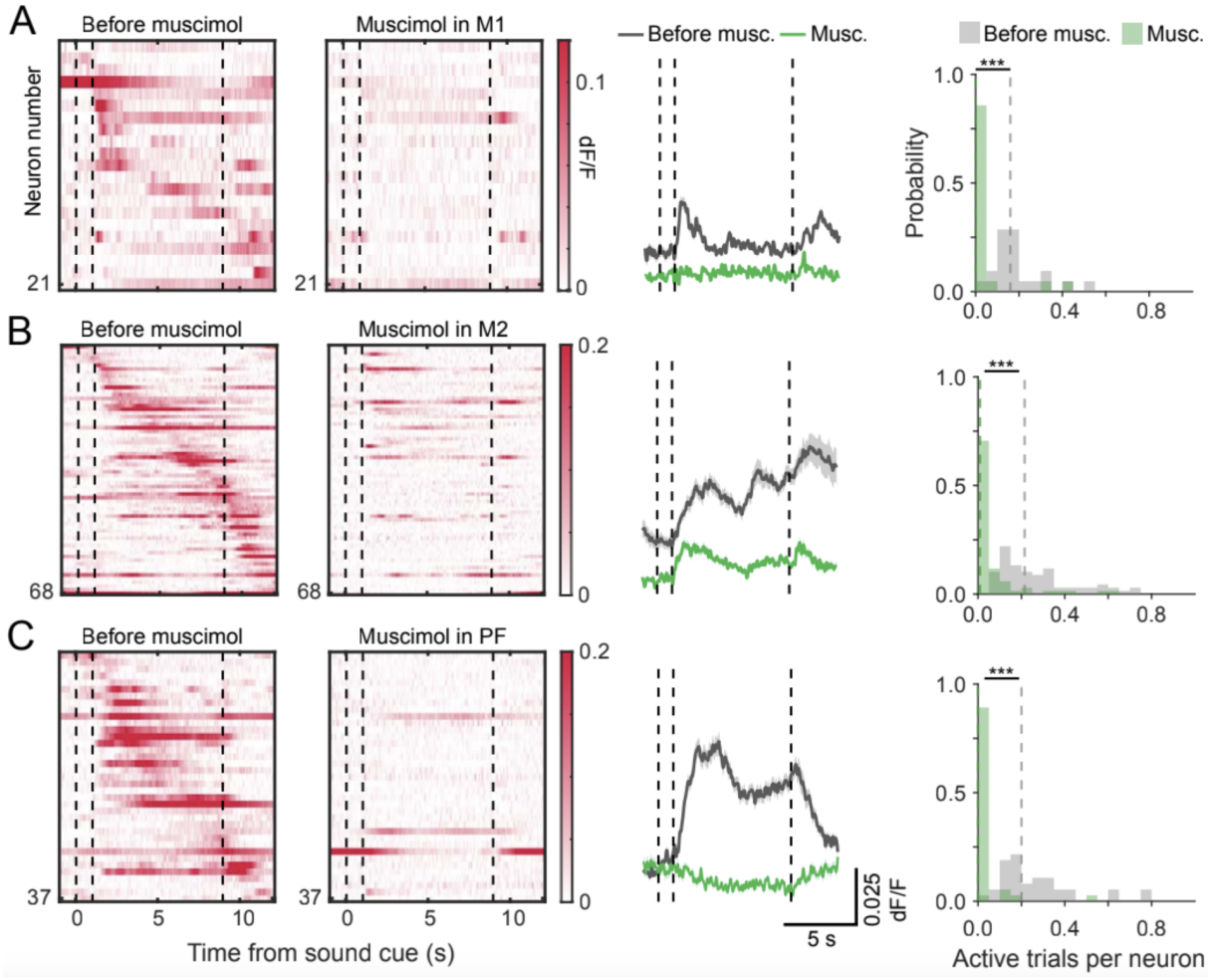


Figure 5.3. Muscimol inactivation in M1, M2, and PF abolishes movement-related striatal activity.

A) From left, trial-averaged activity of MSNs during the ladder task. Dashed lines from left to right in each plot indicate: sound cue onset, ladder onset, and ladder offset; trial-averaged activity of MSNs from the same fields of view, imaged 30 minutes after muscimol injection into M1; average activity of all DLS neurons before and after muscimol injection (mean ± SEM); fraction of active trials in the imaged neurons before and after muscimol injection (n = 21 neurons, of which 14 dMSNs from n = 1 mouse and 7 iMSNs from n = 1 mouse). **B)** Same as A) but for muscimol injections in M2 (n = 68 neurons, of which 24 dMSNs from 3 mice and 44 iMSNs from 4 mice). **C)** Same as A-B) but for muscimol injections in PF (n = 37 neurons, of which 12 dMSNs from 2 mice and 25 iMSNs from 3 mice). Dotted lines indicate medians. Paired Wilcoxon rank-sum test, p ¡ 0.001 for all comparisons after vs. before manipulation.

The average population neuronal activity differed between the four regions (Fig. 5.1D,E). PF neurons were strongly active throughout the movement period. In contrast, M1 and M2 neurons showed more pronounced activity around the onset and offset of ladder movement. MSN activity appeared to be a combination of the CTX and PF patterns, with both sustained activity during movement as well as activity at ladder onset and offset. These results suggest a complementary contribution of cortical and thalamic inputs to striatal MSN activity during movements.

The population average activity of dMSNs and iMSNs exhibited different patterns. dMSN activity peaked at the ladder onset and declined during the movement before peaking again at the ladder offset, while iMSNs exhibited a consistent increase in activity during movement. In the input areas, the population average activity patterns among dMSN- and iMSN-projecting were similar.

5.4.2 Cell-type classification based on ensembles of single-trial neuronal activity

To explore how different the activity of single neurons is between dMSNs and iMSNs, and between dMSN- and iMSN-projecting neurons in each input region, we trained a classifier that predicts the cell type based on the neuron's activity. We designed the Trial Ensemble Attention Network (TEA-net) (Fig. 5.4A) to capture the nonlinearity and temporal structure of neuronal activity, as well as the variability of a neuron's activity across single trials in a fixed-length task. The classifier model is based on an encoder–decoder framework using recurrent neural networks (RNNs) with an attention-based mechanism. The input to the classifier is the neural activity from 2 seconds before to 4 seconds after the ladder movement period, organized into ensembles of a fixed number of randomly selected trials (Fig. 5.4A).

The classifier was trained using 10-fold cross-validation across sessions, such that each session was used in the test set once for each neuron. Using the TEA-net classifier trained on ensembles of trial activity, we found that dMSNs and iMSNs, as well as dMSN- and

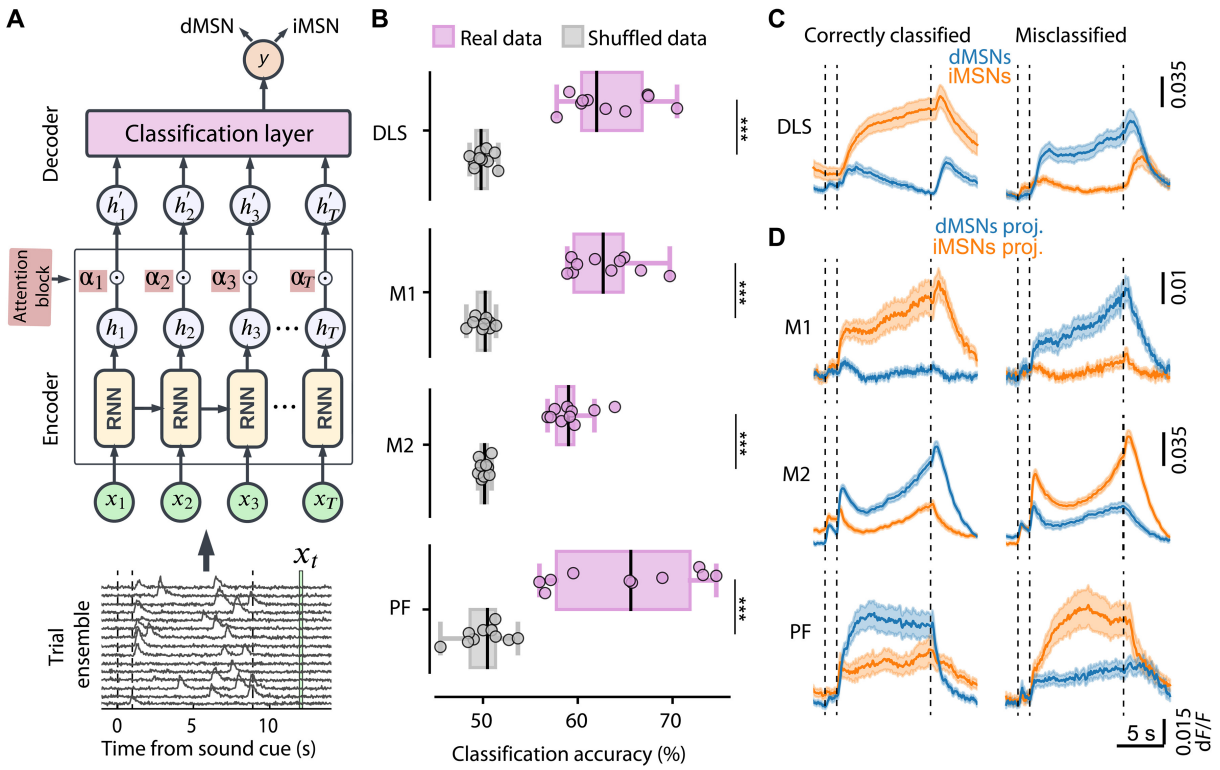


Figure 5.4. TEA-net architecture and cell-type classification performance.
A) Top, schematic of the TEA-net classifier. The encoder consists of RNN units that learn a hidden-state multivariate time-series representation h_t, which is pointwise weighted by attention values a_t and summed to produce the final latent representation. This representation is fed into a fully connected decoder for classification. Bottom, classifier input based on subsampled ensembles of a fixed number of trials. **B)** Cell-type classification accuracy in each region compared to shuffled data. **C)** Average activity of correctly classified and misclassified neurons in DLS. **D)** Average activity of correctly classified and misclassified neurons in M1, M2, and PF. Mean ± SEM shown.

iMSN-projecting neurons, are distinguishable above chance level, although the classification performance was generally modest (Fig. 5.4B,C; Wilcoxon rank-sum test, $p < 0.001$ for all comparisons vs. shuffled data). Classification accuracy was not statistically different across the four regions, suggesting a similar degree of distinction between target-cell types and between their input-cell types in motor cortex and thalamus (Wilcoxon rank-sum test with post-hoc Bonferroni correction for n = 6 comparisons).

We compared the performance of TEA-net with linear and nonlinear baseline classifiers trained on either trial-averaged neural activity or ensembles of single trials (Table 5.1). Classifi-

cation accuracy was highest for TEA-net, which significantly outperformed linear SVMs trained on trial-averaged activity and trial ensembles, as well as a fully connected neural network with attention (Wilcoxon rank-sum test: TEA-net vs. SVM on trial-averaged data, p = 0.004; TEA-net vs. SVM on ensemble data, p ¡ 0.001; TEA-net vs. FCNN, p = 0.011).

Table 5.1. Classification accuracy of dMSNs and iMSNs obtained by training different classifier models with trial-averaged neural activity or random ensembles of single trials. Classification accuracy was highest for TEA-net. Wilcoxon rank-sum test, TEA-net vs. SVM (trial-averaged data) p = 0.004; TEA-net vs. SVM (ensemble data) $p < 0.001$; TEA-net vs. FCNN p = 0.011. SVM: Support Vector Machine; FCNN: fully connected neural network; RNN: recurrent neural network; TEA-net: Trial Ensemble Attention Network.

Model input	**Classifier model**	**Accuracy $\pm$ SD (%)**
Trial-averaged neural activity	Linear SVM	53.87 ± 5.16
Ensembles of trials	Linear SVM	52.31 ± 6.53
Ensembles of trials	FCNN with attention	57.33 ± 3.86
Ensembles of trials	RNN with attention (TEA-net)	63.21 ± 3.98

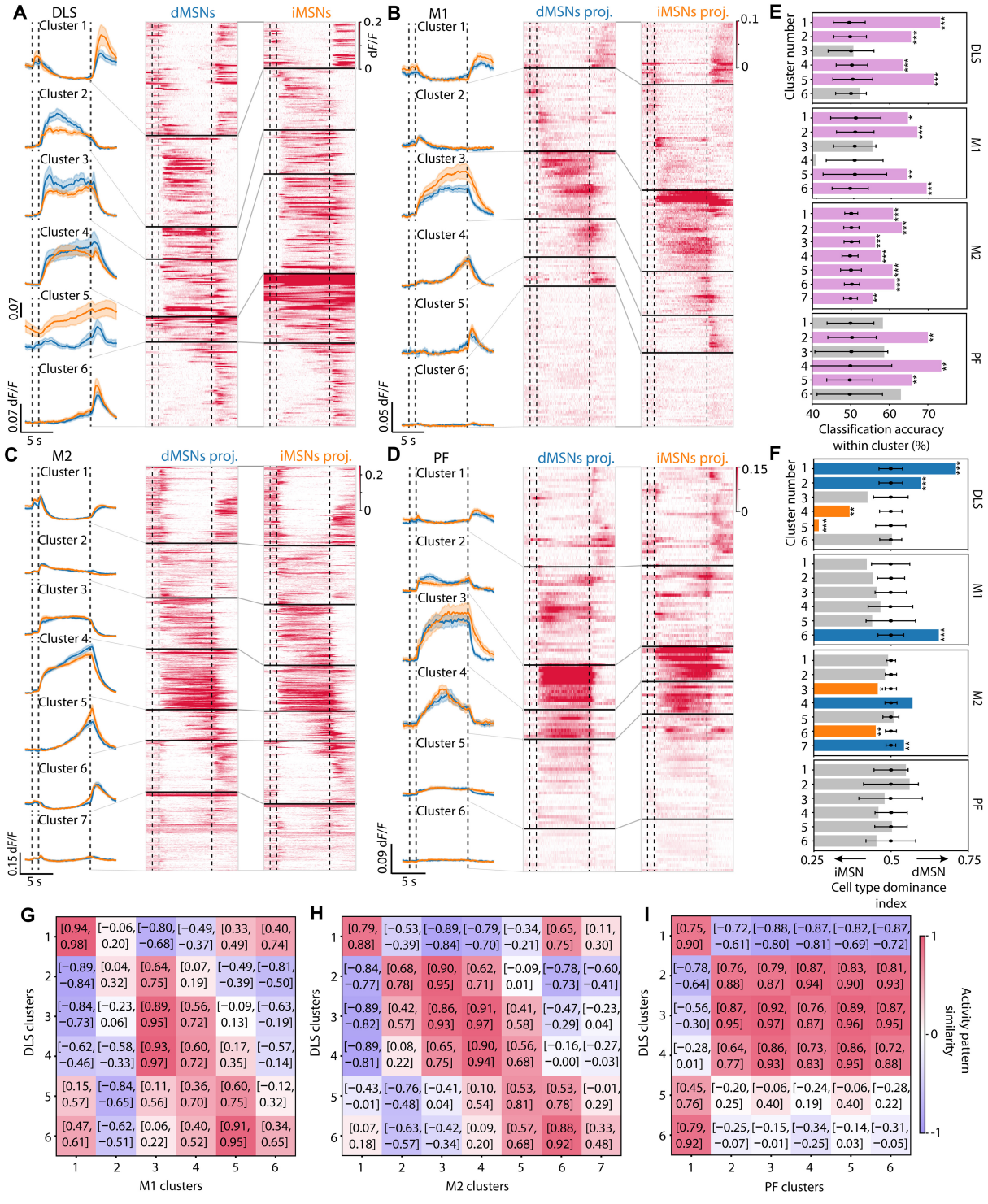
We next examined the mean activity of correctly classified neurons (Fig. 5.4D,E). Correctly classified neurons exhibited stronger differences across cell types compared to misclassified neurons. In DLS, correctly classified dMSNs showed higher activity levels that culminated near the end of movement compared to iMSNs (Fig. 5.4D). Correctly classified iMSN-projecting neurons in M1 exhibited higher activity levels throughout the movement period (Fig. 5.4E), whereas correctly classified dMSN-projecting neurons in M2 and PF showed stronger movement-related activity. The activity patterns of misclassified neurons resembled those of correctly classified neurons of the opposite cell type. Together, these results indicate that neuronal activity patterns contain cell-type-specific information, but that substantial fractions of neurons display activity patterns typical of the other cell type.

5.4.3 Heterogeneous and distinctive activity patterns in DLS and input areas

To explore the heterogeneity of activity patterns and identify those that are distinctive of each cell type, we performed hierarchical clustering of the latent space representations of the

TEA-net model (i.e., the inputs to the classifier). We used the latent space representations as the classifier has been trained to capture separation between cell-types, and this representation has a higher tendency to cluster than raw activity (Additional Fig. 5.6A). We applied a Dynamic Tree Cut algorithm [Langfelder et al., 2008] to the hierarchical clustering dendrogram together with subsampling (Methods, Additional Fig. 5.6B,C) to determine the optimal number of clusters.

Figure 5.5. Activity pattern characterization of MSNs and inputs.
A) From left, population average activity of dMSNs (blue) and iMSNs (orange), and trial-averaged activity of individual dMSNs and iMSNs (mean ± SEM) in each cluster. Neuron ordering follows the leaf sequence of the clustering dendrogram. *n* for dMSNs and iMSNs: cluster 1: 94 and 47; cluster 2: 76 and 63; cluster 3: 27 and 45; cluster 4: 48 and 102; cluster 5: 21 and 71; cluster 6: 71 and 86. **B–D)** Same as A), for dMSN- and iMSN-projecting neurons in M1, M2, and PF. *n* for dMSN-projecting and iMSN-projecting neurons: M1, cluster 1: 20 and 38; cluster 2: 37 and 65; cluster 3: 30 and 50; cluster 4: 17 and 27; cluster 5: 13 and 23; cluster 6: 63 and 46; M2, cluster 1: 352 and 585; cluster 2: 249 and 431; cluster 3: 234 and 445; cluster 4: 277 and 335; cluster 5: 141 and 218; cluster 6: 236 and 461; cluster 7: 360 and 487; PF, cluster 1: 33 and 36; cluster 2: 33 and 30; cluster 3: 15 and 13; cluster 4: 10 and 12; cluster 5: 30 and 39; cluster 6: 15 and 20. **E)** Within-cluster cell-type classification accuracy. Purple bars indicate values significantly above chance accuracy (black circles and error bars, mean ± SD). **F)** Within-cluster cell-type dominance. Values above or below 0.5 indicate dMSN (dMSN-projecting) or iMSN (iMSN-projecting) dominance, respectively. Blue and orange bars indicate significant dMSN (dMSN-projecting) and iMSN (iMSN-projecting) dominance above chance (black circles and error bars, mean ± SD). **G–I)** Activity pattern correlation between DLS and M1, M2, and PF clusters. Numbers indicate 95% confidence intervals of mean correlation values. Dashed lines from left to right indicate sound cue, ladder onset, and offset. $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$.


A
DLS
dMSNs
iMSNs
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
0.2
dF/F
0
0.07
0.07 dF/F
5 s
B
M1
dMSNs proj.
iMSNs proj.
0.1
0
0.05 dF/F
5 s
C
M2
dMSNs proj.
iMSNs proj.
Cluster 7
0.15 dF/F
5 s
D
PF
dMSNs proj.
iMSNs proj.
0.15
0
0.09 dF/F
5 s
E
Cluster number
DLS
M1
M2
PF
Classification accuracy within cluster (%)
F
Cell type dominance index
iMSN
dMSN
G
DLS clusters
M1 clusters
H
DLS clusters
M2 clusters
I
DLS clusters
PF clusters
Activity pattern similarity

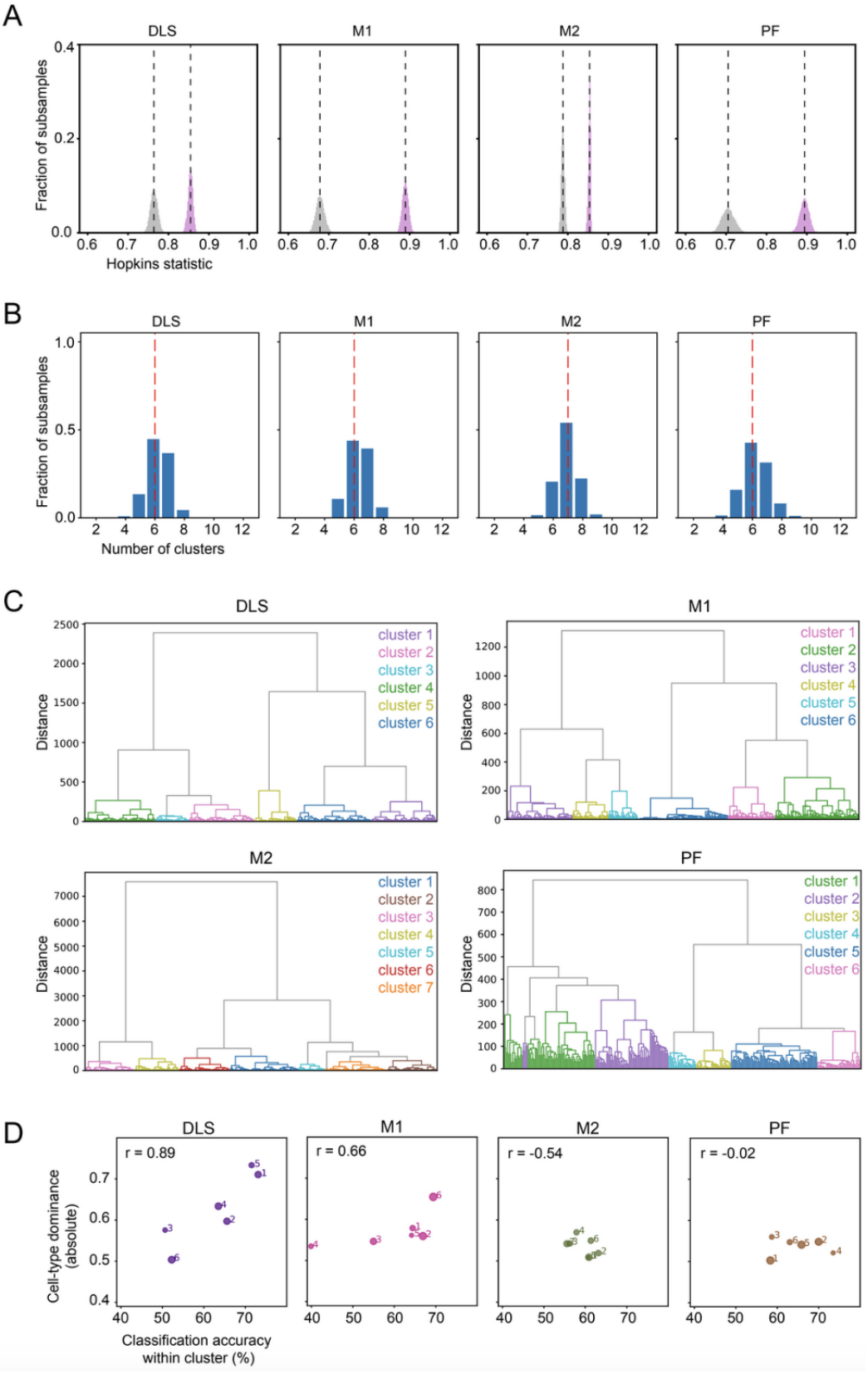
For each neuron that belongs to a specific latent space representation cluster, we plotted the corresponding trial-averaged neuronal activity (Fig. 5.5). We identified several major clusters in each region indicating that functionally heterogeneous subpopulations exist (Fig. 5.5A–D). Examining the classifier performance of neurons belonging to each cluster, we found that some clusters showed much higher classification accuracies than the population mean accuracy (Fig. 5.5E). In many of these clusters with high classification accuracy, one of the two cell-types dominated the cluster, indicating that these activity patterns were found primarily in one of the two cell-types (e.g., dMSN-dominant, DLS clusters 1 and 2; iMSN-dominant, DLS clusters 4 and 5). However, in other clusters with high classification accuracies, cell-type dominance was not apparent (e.g., PF clusters 2 and 4) (Additional Fig. 5.6D), suggesting that these clusters are further heterogeneous within them and additional latent single-trial features might contribute to classifier performance.

DLS clusters enriched in dMSNs showed strong activity near the onset and offset of movement (Fig. 5.5A,F, clusters 1 and 2) while iMSN-dominant clusters (clusters 4–5) exhibited peak activity toward the later phase of movement. In M1 and M2, many clusters were formed by neurons that were active around the onset or offset of the movement (Fig. 5.5B,C; M1 clusters 1, 2, 4 and 5; M2 clusters 1–2 and 5–7). Additionally, in M2, about a third of the neurons belonged to movement-active clusters, with the cluster with consistent activity throughout the movement containing more iMSN-projecting neurons (Cluster 3), and the cluster with activity ramping towards the end of the movement dominated by dMSN-projecting neurons (Cluster 4). Furthermore, a larger fraction of dMSN-projecting M1 and M2 neurons did not show clear activity time-locked to our task (M1 cluster 6 and M2 cluster 7), raising the possibility that these neurons may relay information selectively to dMSNs in behavioral contexts that are not explored in our study. In contrast to cortical inputs, the majority of PF clusters showed activity during the movement (Fig. 5.5D, clusters 2–5).

The results indicate that dMSNs and iMSNs and their projecting neurons display specific movement-related activity patterns, some of which are more common in one cell-type than the

other. We assessed the similarity between DLS clusters and input neuron clusters (Fig. 5.5G–J) by computing Pearson's correlation between their cluster-averaged activity patterns. M1 and M2 clusters more closely resembled movement onset/offset clusters in DLS, as compared to PF. The M1 movement cluster was more similar to late-movement clusters in DLS, while all PF and M2 movement clusters were highly similar to all DLS movement clusters. These results suggest that corticostriatal neurons might play an important role in driving striatal activity at movement start and end, which are more prominent in dMSNs. Thalamostriatal neurons may be more important in driving striatal activity during movements.

Figure 5.6. Latent representations exhibit stronger clustering tendency than raw neural activity.
A) Hopkins statistic distributions of trial-averaged neural activity (in gray) and of the latent space representations (in violet) obtained from 10000 random subsamplings of 80% of the neurons in DLS, M1, M2 and PF. The latent representations yield consistently higher Hopkins values than the corresponding neural activity traces across all four regions, indicating that the latent space possesses a greater likelihood to form clusters. Hopkins statistic for latent representation and neural activity in each region, respectively (mean $\pm$ SD): DLS, 0.85 ± 0.01, 0.76 ± 0.01; M1, 0.89 ± 0.01, 0.68 ± 0.01; M2, 0.85 ± 0.002, 0.79 ± 0.003; PF, 0.89 ± 0.01, 0.71 ± 0.02. Dotted lines indicate the median of the distributions. **B)** Distributions of the number of clusters for the latent space representations, obtained from 10000 random subsamplings of 80% of the neurons. The mode of the distributions coincides with the number of clusters obtained when applying the Dynamic Tree Cut approach to the complete datasets in DLS, M1, M2 and PF (red dotted line). **C)** Dendrograms obtained from the hierarchical clustering of the latent space representations of DLS, M1, M2 and PF neurons. Colors indicate the different clusters of neurons obtained in each region (see Fig. 5.5). **D)** Correlation coefficients between cell-type dominance and classification accuracy in each cluster, for each region. Circle size in each plot corresponds to the cluster size.


A
DLS
M1
M2
PF
Fraction of subsamples
Hopkins statistic
B
Number of clusters
C
Distance
cluster 1
cluster 2
cluster 3
cluster 4
cluster 5
cluster 6
cluster 7
D
r = 0.89
r = 0.66
r = -0.54
r = -0.02
Cell-type dominance (absolute)
Classification accuracy within cluster (%)

5.4.4 Forelimb-related rhythmicity is input- and cell-type specific

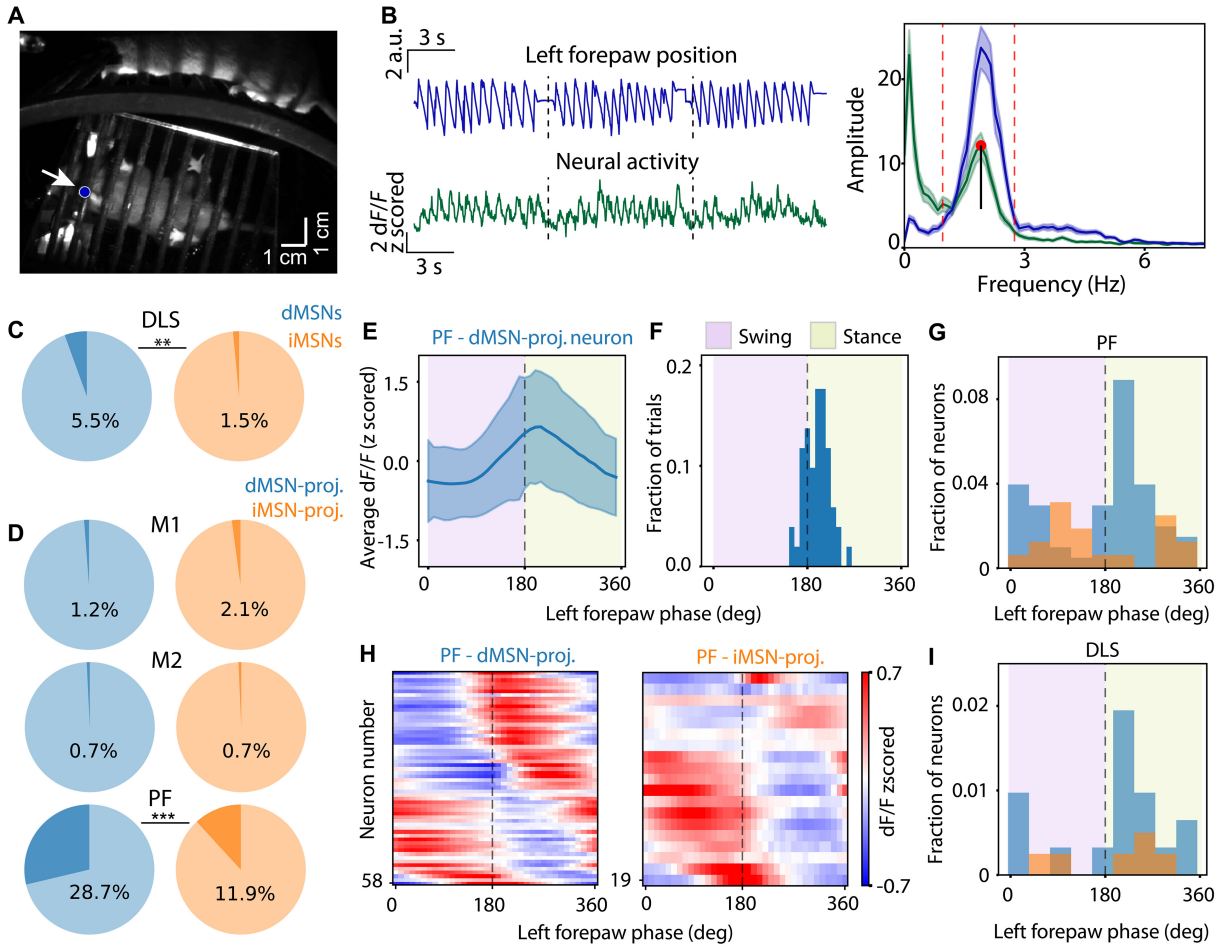
To explore how the neural activity of the two MSN cell types and their specific cortical and thalamic inputs relates to limb kinematics in this motor task, we extracted the time series of the coordinates of the four paws using DeepLabCut Mathis et al. [2018] (Fig. 5.7A). To detect neurons that show rhythmic activity with similar periodicity as the contralateral left forepaw during locomotion, we performed a Fourier analysis on the left forelimb trajectory and on neural activity during the movement period of each trial (Methods; Fig. 5.7B). We found that a small fraction of DLS neurons showed activity matching the rhythmicity of the forelimb, with a higher fraction of rhythmic cells found in dMSNs than in iMSNs (Fig. 5.7C; dMSN rhythmic cells $n = 17$ out of 308 (5.5%), iMSN rhythmic cells $n = 6$ out of 400 (1.5%); Fisher exact test $p_{\text{raw}} = 0.004$, Bonferroni-adjusted $p = 0.018$).

Among the input regions, very few rhythmic cells were observed in M1 and M2 (Fig. 5.7D; M1: dMSN-proj. rhythmic cells $n = 2$ out of 166 (1.2%), iMSN-proj. rhythmic cells $n = 5$ out of 234 (2.1%); M2: dMSN-proj. rhythmic cells $n = 14$ out of 1881 (0.7%), iMSN-proj. rhythmic cells $n = 20$ out of 2865 (0.7%)). In contrast, we found a substantial proportion of rhythmic cells in PF neurons, with dMSN-projecting thalamic neurons containing a significantly higher fraction of rhythmic cells (dMSN-proj. rhythmic cells $n = 58$ out of 202 (28.7%), iMSN-proj. rhythmic cells $n = 19$ out of 160 (11.9%); Fisher exact test $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$).

Each stride in the gait cycle consists of a stance phase, when the forelimb is in contact with the ladder bar, followed by a swing phase, when the forelimb is raised and pushed forward. We examined whether rhythmic neurons are preferentially tuned to a specific phase of left forelimb position (Fig. 5.7E). We found that thalamic dMSN- and iMSN-projecting neurons exhibited opposite phase tuning, with dMSN-projecting neuron activity peaking during the stance phase and iMSN-projecting neuron activity peaking during the swing phase (Fig. 5.7F; Watson two-sample U^2 test, $p = 0.011$). dMSNs in DLS showed a similar forelimb phase

preference to thalamic dMSN-projecting neurons. These findings suggest that the contralateral forelimb position during movement is primarily encoded by thalamostriatal neurons rather than by corticostriatal neurons. Furthermore, thalamostriatal neurons may provide information related to forelimb position to striatal neurons via target cell-type-specific connections.

Figure 5.7. Forelimb-related rhythmic neural activity in striatal and thalamic neurons. **A)** Top, example video frame of a mouse on the motorized ladder and the marked limbs (mirror view) used for training DeepLabCut. Bottom, coordinates of the tracked limbs of a mouse during one imaging session. Top, in dark blue, example trace of the first PCA component of the x coordinate of the left forelimb during three consecutive trials in one imaging session (from ladder onset to 1 s after the ladder offset). In green, corresponding activity trace of a PF neuron recorded during the same imaging session. Bottom, power spectral densities of the forelimb trajectory and neural activity, averaged across all trials of the same imaging session. The dotted lines indicate the full width at 10% maximum of the peak frequency of the forelimb trajectory. Neural activity that peaks within this frequency interval is defined as forelimb-related rhythmic. **C)** Fraction of dMSNs and iMSNs with forelimb-related rhythmic activity (dMSN rhythmic neurons $n = 17$ out of 308, iMSN rhythmic neurons $n = 6$ out of 400). Fisher exact test $p_{\text{raw}} = 0.004$, Bonferroni-adjusted $p = 0.018$. **D)** Fraction of dMSN- and iMSN-projecting neurons with forelimb-related rhythmic activity in M1, M2 and PF. M1: dMSN-proj. rhythmic neurons $n = 2$ out of 166, iMSN-proj. rhythmic neurons $n = 5$ out of 234, Fisher exact test $p_{\text{raw}} = 0.704$, Bonferroni-adjusted $p = 1$; M2: dMSN-proj. rhythmic neurons $n = 14$ out of 1881, iMSN-proj. rhythmic neurons $n = 20$ out of 2865, $p_{\text{raw}} = 0.862$, Bonferroni-adjusted $p = 1$; PF: dMSN-proj. rhythmic neurons $n = 58$ out of 202, iMSN-proj. rhythmic neurons $n = 19$ out of 160, $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$. Comparisons between regions: Fisher exact test, DLS vs. M1 $p_{\text{raw}} = 0.178$, Bonferroni-adjusted $p = 1$; DLS vs. M2 $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$; DLS vs. PF $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$; PF vs. M1 $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$; PF vs. M2 $p_{\text{raw}} < 0.001$, Bonferroni-adjusted $p < 0.001$; M1 vs. M2 $p_{\text{raw}} = 0.037$, Bonferroni-adjusted $p = 0.219$. **E)** Trial-averaged activity (mean $\pm$ SD) of an example dMSN-projecting neuron in PF that peaks at the beginning of the stance phase. F) Histogram of the neural activity peak relative to the left forelimb phase, at each trial for the same neuron (n = 51 trials). G) Histogram of the left forelimb-position phase tuning of rhythmic dMSN- and iMSN-proj. neurons in PF. H) Trial-averaged activity heatmaps of all PF rhythmic dMSN- and iMSN- proj. neurons aligned to the left forepaw phase I) As in G, for dMSNs and iMSNs in DLS.


A
1 cm
1 cm
B
2 a.u.
3 s
Left forepaw position
Neural activity
2 dF/F zscored
3 s
Amplitude
20
10
0
0
3
6
Frequency (Hz)
C
DLS
**
dMSNs
iMSNs
5.5%
1.5%
D
dMSN-proj.
iMSN-proj.
M1
1.2%
2.1%
M2
0.7%
0.7%
PF
***
28.7%
11.9%
E
PF - dMSN-proj. neuron
Average dF/F (z scored)
1.5
0.0
-1.5
0
180
360
Left forepaw phase (deg)
F
Swing
Stance
Fraction of trials
0.2
0.1
0.0
0
180
360
G
PF
Fraction of neurons
0.08
0.04
0
0
180
360
Left forepaw phase (deg)
H
PF - dMSN-proj.
PF - iMSN-proj.
Neuron number
58
19
0
180
360
0.7
-0.7
dF/F zscored
Left forepaw phase (deg)
I
DLS
Fraction of neurons
0.02
0.01
0
0
180
360
Left forepaw phase (deg)

5.5 Discussion

This study characterized the activity dynamics and kinematic-related information of pre- and postsynaptic populations of four motor areas. By imaging DLS cell types and their cortical and thalamic presynaptic inputs, we aimed to identify cell-type specific differences of activity and whether they arise from specificity of presynaptic inputs.

We identified subpopulations with distinct activity patterns within MSNs and within their cortical and thalamic inputs. All neuronal groups (dMSNs, iMSNs, and their respective inputs) showed heterogeneous activity patterns and we did not find any activity pattern that was only observed in one cell type. Nevertheless, dMSNs were predominant in clusters with activity around movement initiation and termination, while more iMSNs were active during movements. This finding aligns with previous studies that have highlighted co-activation of the two cell types but with distinct patterns [Cui et al., 2013, Jin et al., 2014, Meng et al., 2018, Parker et al., 2018, Markowitz et al., 2018, Cruz et al., 2022]. The majority in corticostriatal and thalamostriatal populations showed activity around onset/offset of movement and during movement, respectively. This suggests a complementary contribution of these inputs to striatal activity, with cortical and thalamic activity more closely matching that of dMSNs and iMSNs, respectively. Indeed, even though cortex and thalamus connect with both dMSNs and iMSNs, it has been suggested that cortical and thalamic inputs are biased towards dMSNs and iMSNs, respectively [Nelson et al., 2021, Kress et al., 2013, Parker et al., 2016, Johansson and Silberberg, 2020, Zhang et al., 2025], although an opposite observation has also been made for the thalamostriatal connectivity [Parker et al., 2016]. These observations together suggest that corticostriatal and thalamostriatal connections have distinct functions, with corticostriatal neurons driving striatal activity primarily at the onset and offset of movements, while thalamostriatal neurons contribute more to striatal activity during the movements. These findings suggest that striatal activity during movements arises as an integration of both corticostriatal and thalamostriatal inputs, instead of being primarily driven by cortical activity [Lemke et al., 2019, Peters et al., 2021].

Our clustering analysis did not reveal clear differences in dMSN- and or iMSN-projecting neurons that correspond to differences in dMSNs and iMSNs. However, in a separate analysis, we found movement kinematics-related neuronal rhythmicity primarily in PF but also in DLS. Importantly, a majority of these neurons in PF were dMSN-projecting neurons, and a majority in DLS were dMSNs with phase tuning matching that of dMSN-projecting PF neurons. Taken together, the findings suggest that while dMSN- and iMSN-projecting neurons in each region might share many activity patterns (and some are likely overlapping populations), a subset of them encode and transmit kinematic or task-related information in a cell-type specific manner. Future studies should further test this idea with simultaneous recordings of identified pre- and post-synaptic populations as well as pathway- and cell-type-specific activity manipulations. Furthermore, kinematic encoding in the PF-DLS pathway should be further studied to determine if it is restricted to rhythmic locomotion studied here or more broadly observed in other movement contexts. We note that studies with electrophysiological recordings have described a higher fraction of locomotion-related rhythmic cells both in the DLS as well as in M1 compared to what we report here [Yang et al., 2024, Singla et al., 2025]. Our study using calcium imaging may be underestimating rhythmic activity.

Machine learning-based methods are becoming increasingly popular for identifying cell types based on neuronal activity [Schneider et al., 2023a, Ophir et al., 2024, Lee et al., 2024, Beau et al., 2025]. In this study, we used a supervised classification approach based on recurrent neural networks as a tool for determining functional differences among cell types. Our classifier design combined trial ensemble inputs and nonlinear temporal modeling via the RNN, and outperformed baseline models under cross-validation. By incorporating neuronal activity across multiple single trials, rather than using trial-averaged activity the classifier may capture features—such as fraction of responsive trials, response amplitude and timing, and their consistency across trials—that determine how reliably and strongly information is encoded and transmitted within circuits. These features may prove crucial for both determining specificity and connectivity rules in cell-type defined circuits. While our design accounts for the nonlinearity

of encoding, temporal structure of neuronal activity, and single-trial activity, it is aligned to task events and does not account for trial-by-trial variability in the movements of the animal. Incorporating kinematic features, e.g., paw position and velocity, in future designs would have the potential to enhance the quantification of functional diversity of neuronal populations in relation to the generation of movements and offer mechanistic insights into how cell-type specific circuits generate precise movements.

Acknowledgements

Chapter 5, in full, is a reprint of the following material: Gjoni, Enida*; Sristi, Ram Dyuthi*; Liu, Haixin*; Dror, Shahar; Lin, Xinlei; O'Neil, Keelin; Arroyo, Oscar M.; Hong, Sun Woo; Kim, Hannah; Liu, Jeffrey; Blumenstock, Sonja; Lim, Byungkook; Mishne, Gal†; Komiyama, Takaki† (2026). "Complementary Cortical and Thalamic Contributions to Cell Type–Specific Striatal Activity Dynamics During Movement." *Science Advances*, 12(5): eaea3935. * denotes co–first authors; # denotes co–corresponding authors. The dissertation author contributed equally to this work and co-led conceptualization, methodology development, formal analysis, and writing of the manuscript. The dissertation author led the computational modeling and analysis of neural data and contributed to the interpretation of results in close collaboration with experimental co-authors.

Chapter 6

Discussion and Future Directions

6.1 Discussion

This dissertation develops interpretable machine learning frameworks for understanding distributed and context-dependent neural computation from high-dimensional biological data. Across the methodological contributions and the neuroscience application presented in earlier chapters, a central theme emerges: meaningful biological interpretation requires explicitly modeling structure that lies between purely global descriptions and purely local feature-level analyses. Neural and biological systems exhibit shared dynamics across regions and conditions, alongside context-specific and pathway-specific variations, and the methods developed in this thesis are designed to disentangle these complementary sources of variability.

A unifying principle across all contributions is the emphasis on structure-aware interpretability. Rather than ranking individual features or neurons in isolation, the proposed frameworks identify structured sets of features, latent subspaces, or neural populations that jointly encode behavior or condition differences. This perspective aligns more closely with biological organization, where coordinated activity patterns, rather than single units, form the basis of computation. By focusing on groups of features and population-level representations, the methods in this dissertation provide interpretations that are both more stable and more biologically meaningful.

Another shared theme is the explicit separation of shared and specific information. In

the coupled transformer autoencoder, this separation is achieved through orthogonal shared and private latent subspaces that disentangle common dynamics across brain regions from region-specific activity. In conditional stochastic gates, feature relevance is conditioned on contextual variables, allowing the model to distinguish context-invariant features from those that are selectively informative under specific conditions. In differential spectral clustering, the focus shifts to identifying feature groups whose connectivity structure differs across conditions, separating shared correlation structure from condition-specific interactions. Together, these approaches highlight the importance of modeling heterogeneity across contexts, regions, and conditions rather than collapsing all data into a single representation.

A further unifying aspect of this work is its population-level focus. Across methods, the goal is not to explain individual neurons or features in isolation, but to capture coordinated patterns of activity and structured feature groups that better reflect the organization of biological systems. This population-level reasoning is particularly important in modern neuroscience and biomedical datasets, where high dimensionality and strong dependencies between variables are the norm. The frameworks developed here are designed to operate in this regime, providing interpretable outputs that remain meaningful even as data dimensionality scales.

Finally, the methods introduced in this dissertation are designed to be model-agnostic in their downstream use. The latent representations, selected feature sets, and differential clusters produced by these frameworks can be readily integrated into decoding, clustering, or hypothesis-generation pipelines without tying interpretation to a single predictive model. This flexibility makes the approaches broadly applicable across domains and encourages their use as exploratory tools for scientific discovery.

6.2 Contributions

Differential spectral clustering reframes feature selection as a graph comparison problem, focusing on differences in inter-feature connectivity rather than individual feature values. By

constructing condition-specific graphs and identifying groups of features whose connectivity patterns differ across conditions, DiSC captures differential structure that may be invisible to methods based on sparsity or marginal relevance. Theoretical analysis under structured random graph models provides insight into when such differential groups can be reliably recovered, while experiments across synthetic data, imaging, hyperspectral analysis, and fMRI demonstrate the practical utility of this approach. DiSC complements context-aware feature selection by addressing a distinct notion of difference, namely changes in correlation structure rather than changes in feature relevance alone.

Conditional stochastic gates challenge the conventional assumption that feature relevance is fixed across all samples. By conditioning feature selection on auxiliary context variables, c-STG models feature importance as a function of context, even when the context does not directly affect the prediction target. This framework reveals that features can be selectively informative under specific conditions, an observation that is particularly relevant in neuroscience and biomedical applications where task demands, internal states, or experimental manipulations modulate neural relevance. Through the use of a hypernetwork that predicts gating parameters, c-STG provides a flexible and scalable mechanism for learning context-dependent relevance patterns, leading to improved predictive performance and enhanced interpretability compared to population-level feature selection methods.

The coupled transformer autoencoder addresses a fundamental challenge in systems neuroscience: how to jointly model neural activity recorded simultaneously across multiple brain regions while preserving both shared dynamics and region-specific structure. By explicitly partitioning each region's latent space into orthogonal shared and private subspaces, the model enables direct investigation of population-level neural dynamics that generalize across regions, while simultaneously isolating region-specific computations. The use of transformer-based sequence models allows CTAE to capture nonlinear and non-stationary temporal dynamics, making it well-suited for large-scale electrophysiology and imaging datasets. Compared to alignment-based or single-region approaches, CTAE yields representations that improve behavioral decoding and

offer clearer interpretability of distributed neural processes.

This dissertation also includes a neuroscience study examining how cortical and thalamic inputs differentially shape cell-type-specific striatal activity during movement. Using large-scale calcium imaging data recorded across multiple brain regions under non-simultaneous experimental conditions, this work focuses on characterizing population-level structure in striatal dynamics when direct joint modeling across regions is not feasible. To this end, Trial Ensemble Attention Networks (TEA-net) are introduced as a representational probe for assessing ensemble-level separability of neural activity patterns across behavioral phases and input pathways. This analysis reveals pronounced functional heterogeneity within both direct- and indirect-pathway striatal neurons, as well as complementary contributions of cortical and thalamic inputs to movement initiation and execution. By highlighting cell-type-specific and region-dependent structure under realistic experimental constraints, this study provides insight into how distributed motor signals are integrated within the striatum and motivates the need for interpretable, population-level approaches to neural data analysis.

6.3 Limitations

While the frameworks developed in this dissertation provide new tools for interpretable analysis of high-dimensional neural data, several limitations remain.

First, the methods rely on hyperparameter choices that can influence both performance and interpretability. In Differential Spectral Clustering (DiSC), the dimensionality of the condition-specific spectral embeddings determines which differential feature groups are recovered; overly large values may capture noise while overly small values may miss meaningful structure. Similarly, Conditional Stochastic Gates require tuning of regularization parameters that control sparsity of the gating mechanism, and Coupled Transformer Autoencoders require specifying the dimensionality of shared and region-specific latent subspaces. Although cross-validation provides a practical solution, principled and automated strategies for selecting these

parameters remain an open methodological challenge.

Second, each method targets a particular notion of structure in the data. DiSC identifies differences in inter-feature connectivity rather than differences in feature magnitudes, and therefore may not detect signals that manifest primarily through changes in mean activity. Conversely, feature-selection approaches such as c-STG focus on identifying predictive features but do not explicitly model relationships between features or coordinated population dynamics. Similarly, latent-variable models such as CTAE capture shared and region-specific temporal dynamics but rely on assumptions about latent subspace structure. Developing unified frameworks that simultaneously model feature relationships, contextual relevance, and multi-region dynamics remains an important open direction.

Third, the current models do not directly incorporate causal perturbations or mechanistic constraints. While the representations and feature groups discovered by these methods can generate biological hypotheses, they do not by themselves establish causal relationships between neural activity and behavior. Integrating such frameworks with perturbation experiments, interventional data, or mechanistic circuit models will be necessary to move from descriptive representations toward causal understanding.

Finally, although the proposed approaches scale well to modern datasets, most experiments presented in this dissertation involve a limited number of experimental conditions or brain regions. As large-scale recording technologies continue to expand to tens of thousands of neurons across many regions and behavioral contexts, further methodological development will be needed to ensure that interpretable representations remain tractable and stable in increasingly large and heterogeneous datasets.

6.4 Future Directions

The methods developed in this dissertation open several promising directions for future research. A natural methodological extension is the development of unified frameworks that

combine the complementary perspectives introduced in this work. Differential spectral clustering focuses on changes in feature connectivity, conditional stochastic gates model context-dependent feature relevance, and coupled autoencoders capture shared and region-specific population dynamics across brain regions. Integrating these ideas into a single framework could enable models that simultaneously identify relevant neural features, characterize how their interactions change across conditions, and represent their shared dynamics across distributed circuits.

Another important direction is improving the adaptability of the proposed models. For example, learning the dimensionality of shared and region-specific latent subspaces directly from data would allow coupled autoencoder models to flexibly adapt to datasets with varying degrees of inter-regional coordination. Similarly, extending differential spectral clustering to continuous task variables or temporal contexts would allow hierarchical analyses of how neural connectivity structure evolves across time or behavioral state.

In neuroscience applications, the increasing availability of simultaneous multi-region recordings provides an opportunity to apply these frameworks to larger-scale circuit analyses. Shared latent representations offer a principled way to compare neural dynamics across animals, experimental conditions, or behavioral tasks, potentially enabling population-level alignment and cross-subject generalization. Such representations may also serve as building blocks for large-scale pretrained models of neural population activity.

More broadly, the principles developed in this dissertation extend beyond neuroscience. Many biomedical datasets, including single-cell transcriptomics, spatial omics, clinical time series, and multimodal physiological recordings, exhibit strong dependencies between variables and context-dependent patterns of relevance. Structure-aware machine learning methods that explicitly disentangle shared and specific sources of variation may therefore provide powerful tools for understanding complex biological systems across domains.

Ultimately, progress in interpreting large-scale biological data will likely require close integration between machine learning, experimental neuroscience, and domain knowledge. The frameworks introduced in this dissertation represent a step toward this goal by demonstrating

how interpretable representation learning can reveal structured organization in complex neural systems.

6.5 Conclusion

Understanding how neural populations give rise to behavior requires analytical frameworks that can reveal structure in high-dimensional, distributed, and context-dependent biological data. This dissertation develops interpretable machine learning methods designed to address this challenge from three complementary perspectives: reorganization across conditions, context-dependent feature relevance, and distributed computation across brain regions.

First, Differential Spectral Clustering (DiSC) demonstrates that meaningful differences between experimental conditions often manifest not only through changes in individual feature activity, but also through changes in the relational structure between features. By identifying groups of features whose connectivity patterns reorganize across conditions, DiSC provides a principled framework for detecting coordinated population-level changes that may remain hidden to traditional feature-wise analyses.

Second, Conditional Stochastic Gates (c-STG) reveal that feature relevance itself can depend on contextual variables, even when those variables do not directly determine the prediction target. By conditioning feature selection on context, this framework exposes selective engagement of features under specific conditions, providing a more flexible and biologically realistic model of relevance in systems where neural responses depend strongly on behavioral or experimental context.

Third, the Coupled Transformer Autoencoder (CTAE) addresses the problem of modeling distributed neural activity across multiple brain regions. By disentangling shared and region-specific latent dynamics, CTAE provides a structured representation of neural computation that captures both global coordination and regional specialization within neural circuits.

Complementing these methodological developments, the neuroscience study presented in

this dissertation demonstrates how such population-level representations can provide insight into real biological systems. By analyzing striatal population activity driven by cortical and thalamic inputs, the work highlights the functional heterogeneity of striatal neurons and illustrates how distributed motor signals are integrated across pathways and behavioral phases.

Taken together, the contributions of this dissertation emphasize that interpretable machine learning for biological systems requires models that explicitly capture structure at multiple levels: relationships between features, conditional relevance across contexts, and coordinated dynamics across brain regions. Approaches that account for these forms of structure provide not only improved predictive performance, but also representations that align more closely with the organization of biological systems.

As experimental technologies continue to scale, enabling recordings across increasingly large neural populations, brain regions, and experimental conditions, the need for such structure-aware analytical frameworks will only grow. The methods developed in this dissertation provide a foundation for future work toward interpretable, scalable, and biologically grounded machine learning approaches for studying complex neural and biomedical systems.

Bibliography

Hamidreza Abbaspourazad, Eray Erturk, Bijan Pesaran, and Maryam M. Shanechi. Dynamical flexible inference of nonlinear latent factors and structures in neural population activity. *Nature Biomedical Engineering*, 8(1):85–108, 2024. doi: 10.1038/s41551-023-01106-1. URL https://doi.org/10.1038/s41551-023-01106-1. Epub 2023-12-11.

Emmanuel Abbe, Afonso S Bandeira, and Georgina Hall. Exact recovery in the stochastic block model. *IEEE Transactions on information theory*, 62(1):471–487, 2015.

Kingma DP Ba J Adam et al. A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 1412(6), 2014.

Maruan Al-Shedivat, Avinava Dubey, and Eric Xing. Contextual explanation networks. *The Journal of Machine Learning Research*, 21(1):7950–7993, 2020.

Genevera I Allen. Automatic feature selection via weighted kernels and regularization. *Journal of Computational and Graphical Statistics*, 22(2):284–299, 2013.

William E Allen, Michael Z Chen, Nandini Pichamoorthy, Rebecca H Tien, Marius Pachitariu, Liqun Luo, and Karl Deisseroth. Thirst regulates motivated behavior through modulation of brainwide neural population dynamics. *Science*, 364(6437):eaav3932, 2019.

Orly Alter, Patrick O Brown, and David Botstein. Singular value decomposition for genome-wide expression data processing and modeling. *Proceedings of the National Academy of Sciences*, 97(18):10101–10106, 2000.

G. Andrew, R. Arora, J. Bilmes, and K. Livescu. Deep canonical correlation analysis. In *Proc. of the 30th International Conference on Machine Learning*, pages 1247–1255, 2013.

Dora Angelaki, Brandon Benson, Julius Benson, Daniel Birman, Niccolò Bonacchi, Kcénia Bougrova, Sebastian A Bruijns, Matteo Carandini, Joana A Catarino, et al. A brain-wide map of neural activity during complex behaviour. *Nature*, 645(8079):177–191, 2025.

Emily W Avery, Kwangsun Yoo, Monica D Rosenberg, Abigail S Greene, Siyuan Gao, Duk L Na, Dustin Scheinost, Todd R Constable, and Marvin M Chun. Distributed patterns of functional

connectivity predict working memory performance in novel healthy and memory-impaired individuals. *Journal of cognitive neuroscience*, 32(2):241–255, 2020.

Muhammed Fatih Balin, Abubakar Abid, and James Zou. Concrete autoencoders: Differentiable feature selection and reconstruction. In *Proceedings of the 36th International Conference on Machine Learning*, pages 444–453. PMLR, 2019.

Linas Baltrunas and Francesco Ricci. Context-based splitting of item ratings in collaborative filtering. In *Proceedings of the third ACM conference on Recommender systems*, pages 245–248, 2009.

Monami Banerjee and Nikhil R Pal. Feature selection with svd entropy: Some modification and extension. *Information Sciences*, 264:118–134, 2014.

Florent Barthas and Alex C Kwan. Secondary motor cortex: where 'sensory'meets 'motor'in the rodent frontal cortex. *Trends in neurosciences*, 40(3):181–193, 2017.

Roberto Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks*, 5(4):537–550, 1994.

Maxime Beau, David J Herzfeld, Francisco Naveros, Marie E Hemelt, Federico D'Agostino, Marlies Oostland, Alvaro Sánchez-López, Young Yoon Chung, Michael Maibach, Stephen Kyranakis, et al. A deep learning strategy to identify cell types across species from high-density extracellular recordings. *Cell*, 188(8):2218–2234, 2025.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in neural information processing systems*, pages 585–591, 2002.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Dimitris Bertsimas, Martin S. Copenhaver, and Rahul Mazumder. The trimmed lasso: Sparsity and robustness, 2017.

Yochai Blau and Tomer Michaeli. Non-redundant spectral dimensionality reduction. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 256–271. Springer, 2017.

Malgorzata Bogdan, Ewout van den Berg, Weijie Su, and Emmanuel Candes. Statistical estimation and testing via the sorted l_1 norm. *arXiv preprint arXiv:1310.1969*, 2013.

Howard D Bondell and Brian J Reich. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with OSCAR. *Biometrics*, 64(1):115–123, 2008.

Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. *Advances in neural information processing systems*, 29, 2016.

Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, 6, 1993.

M Yu Byron, Gopal Santhanam, Maneesh Sahani, and Krishna V Shenoy. Neural decoding for motor and communication prostheses. In *Statistical Signal Processing for Neuroscience and Neurotechnology*, pages 219–263. Elsevier, 2010.

Deng Cai, Chiyuan Zhang, and Xiaofei He. Unsupervised feature selection for multi-cluster data. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 333–342, 2010.

Jianhua Cang and David A Feldheim. Developmental mechanisms of topographic map formation and alignment. *Annual review of neuroscience*, 36(1):51–77, 2013.

R. Caye Daudt, B. Le Saux, A. Boulch, and Y. Gousseau. Urban change detection for multispectral earth observation using convolutional neural networks. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, July 2018.

Jianbo Chen, Mitchell Stern, Martin J Wainwright, and Michael I Jordan. Kernel feature selection via conditional covariance minimization. In *Advances in Neural Information Processing Systems*, pages 6946–6955, 2017.

Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. Learning to explain: An information-theoretic perspective on model interpretation. In *International conference on machine learning*, pages 883–892. PMLR, 2018.

Susu Chen, Yi Liu, Ziyue Aiden Wang, Jennifer Colonell, Liu D Liu, Han Hou, Nai-Wen Tien, Tim Wang, Timothy Harris, Shaul Druckmann, et al. Brain-wide neural activity underlying memory-guided movement. *Cell*, 187(3):676–691, 2024.

Cynthia A Chestek, Vikash Gilja, Paul Nuyujukian, Justin D Foster, Joline M Fan, Matthew T Kaufman, Mark M Churchland, Zuley Rivera-Alvidrez, John P Cunningham, Stephen I Ryu, et al. Long-term stability of neural prosthetic control signals from silicon cortical arrays in rhesus macaque motor cortex. *Journal of neural engineering*, 8(4):045005, 2011.

Mark M Churchland and Krishna V Shenoy. Preparatory activity and the expansive null-space. *Nature Reviews Neuroscience*, 25(4):213–236, 2024.

Mark M Churchland, John P Cunningham, Matthew T Kaufman, Stephen I Ryu, and Krishna V

Shenoy. Cortical preparatory activity: representation of movement or first cog in a dynamical machine? *Neuron*, 68(3):387–400, 2010.

Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487(7405):51–56, 2012.

Andrzej Cichocki, Danilo Mandic, Lieven De Lathauwer, Guoxu Zhou, Qibin Zhao, Cesar Caiafa, and Huy Anh Phan. Tensor decompositions for signal processing applications: From two-way to multiway component analysis. *IEEE signal processing magazine*, 32(2):145–163, 2015.

David Cohen, Tal Shnitzer, Yuval Kluger, and Ronen Talmon. Manifest: Manifold-based feature selection for small data sets, 2022. URL https://arxiv.org/abs/2207.08574.

Ronald R Coifman and Matthew J Hirn. Diffusion maps for changing data. *Applied and computational harmonic analysis*, 36(1):79–107, 2014.

Ronald R Coifman and Stéphane Lafon. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3): 273–297, 1995.

Rui M Costa, Dana Cohen, and Miguel AL Nicolelis. Differential corticostriatal plasticity during fast and slow motor skill learning in mice. *Current Biology*, 14(13):1124–1134, 2004.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Hoboken, NJ, 2 edition, 2005. ISBN 978-0-471-24195-9.

Ian Connick Covert, Wei Qiu, Mingyu Lu, Na Yoon Kim, Nathan J White, and Su-In Lee. Learning to maximize mutual information for dynamic feature selection. In *International Conference on Machine Learning*, pages 6424–6447. PMLR, 2023.

Bruno F Cruz, Gonçalo Guiomar, Sofia Soares, Asma Motiwala, Christian K Machens, and Joseph J Paton. Action suppression reveals opponent parallel control via striatal circuits. *Nature*, 607(7919):521–526, 2022.

Guohong Cui, Sang Beom Jun, Xin Jin, Michael D Pham, Steven S Vogel, David M Lovinger, and Rui M Costa. Concurrent activation of striatal direct and indirect pathways during action initiation. *Nature*, 494(7436):238–242, 2013.

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014a.

John P. Cunningham and Byron M. Yu. Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17(11):1500–1509, 2014b. doi: 10.1038/nn.3776. URL https://doi.org/10.1038/nn.3776.

Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and C Sinan Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 63 (1):1–38, 2010.

D Devakumari and K Thangavel. Unsupervised adaptive floating search feature selection based on contribution entropy. In *2010 international conference on communication and computational intelligence (INCOCCI)*, pages 623–627. IEEE, 2010.

Edgar Díaz-Hernández, Rubén Contreras-López, Asai Sánchez-Fuentes, Luis Rodríguez-Sibrían, Josué O Ramírez-Jarquín, and Fatuel Tecuapetla. The thalamostriatal projections contribute to the initiation and execution of a sequence of movements. *Neuron*, 100(3):739–752, 2018.

Jiang Dong, Dafang Zhuang, Yaohuan Huang, and Jingying Fu. Advances in multi-sensor data fusion: Algorithms and applications. *Sensors*, 9(10):7771–7784, 2009.

Laura N Driscoll, Krishna Shenoy, and David Sussillo. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nature Neuroscience*, 27(7):1349–1363, 2024.

Carmeline J Dsilva, Ronen Talmon, Ronald R Coifman, and Ioannis G Kevrekidis. Parsimonious representation of nonlinear dynamical systems through manifold learning: A chemotaxis case study. *Applied and Computational Harmonic Analysis*, 44(3):759–773, 2018.

Joshua T Dudman and John W Krakauer. The basal ganglia: from motor commands to the control of vigor. *Current opinion in neurobiology*, 37:158–166, 2016.

Michael N Economo, Sarada Viswanathan, Bosiljka Tasic, Erhan Bas, Johan Winnubst, Vilas Menon, Lucas T Graybuck, Thuc Nghi Nguyen, Kimberly A Smith, Zizhen Yao, et al. Distinct descending motor cortex pathways and their roles in movement. *Nature*, 563(7729):79–84, 2018.

Albert Einstein. Zur Elektrodynamik bewegter Körper. *Annalen der Physik*, 322(10):891–921, 1905.

Michael T Eismann, Joseph Meola, and Russell C Hardie. Hyperspectral change detection in the presence of diurnal and seasonal variations. *IEEE Transactions on Geoscience and Remote Sensing*, 46(1):237–249, 2007.

Michael Elad. *Sparse and redundant representations: from theory to applications in signal and*

image processing, volume 2, chapter Prologue, pages 3–15. Springer, 2010.

Gamaleldin F Elsayed, Antonio H Lara, Matthew T Kaufman, Mark M Churchland, and John P Cunningham. Reorganization between preparatory and movement population responses in motor cortex. *Nature communications*, 7(1):13239, 2016.

Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. Normalized mutual information feature selection. *IEEE Transactions on Neural Networks*, 20(2):189–201, 2009.

Jonas Everaert, Hadas Benisty, Reuma Gadassi Polack, Jutta Joormann, and Gal Mishne. Which features of repetitive negative thinking and positive reappraisal predict depression? an in-depth investigation using artificial neural networks with feature selection. *Journal of Psychopathology and Clinical Science*, 2022.

Isabella P Fallon, Ryan N Hughes, Francesco Paulo Ulloa Severino, Namsoo Kim, Clara M Lawry, Glenn DR Watson, Marina Roshchina, and Henry H Yin. The role of the parafascicular thalamic nucleus in action initiation and steering. *Current Biology*, 33(14):2941–2951, 2023.

Gidon Felsen and Zachary F Mainen. Neural substrates of sensory-guided locomotor decisions in the rat superior colliculus. *Neuron*, 60(1):137–148, 2008.

Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 7–16, 2014.

J. Feng and N. Simon. Sparse-Input Neural Networks for High-dimensional Nonparametric Regression and Classification. *ArXiv e-prints*, November 2017.

Mario Figueiredo and Robert Nowak. Ordered weighted l_1 regularized regression with strongly correlated covariates: Theoretical aspects. In *Artificial Intelligence and Statistics*, pages 930–938. PMLR, 2016.

Mikhail Figurnov, Shakir Mohamed, and Andriy Mnih. Implicit reparameterization gradients. In *Advances in Neural Information Processing Systems*, pages 441–452, 2018.

Emily S Finn, Xilin Shen, Dustin Scheinost, Monica D Rosenberg, Jessica Huang, Marvin M Chun, Xenophon Papademetris, and R Todd Constable. Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity. *Nature neuroscience*, 18(11): 1664–1671, 2015.

Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

Keinosuke Fukunaga and Warren LG Koontz. Application of the Karhunen-Loeve expansion to

feature selection and ordering. *IEEE Transactions on computers*, 100(4):311–318, 1970.

Stefano Fusi, Earl K Miller, and Mattia Rigotti. Why neurons mix: high dimensionality for higher cognition. *Current opinion in neurobiology*, 37:66–74, 2016.

Siyuan Gao, Gal Mishne, and Dustin Scheinost. A hierarchical manifold learning framework for high-dimensional neuroimaging data. In *International Conference on Information Processing in Medical Imaging*, pages 631–643. Springer, 2019.

Siyuan Gao, Gal Mishne, and Dustin Scheinost. Nonlinear manifold learning in functional magnetic resonance imaging uncovers a low-dimensional space of brain dynamics. *Human brain mapping*, 42(14):4510–4524, 2021.

Claire E Geddes, Hao Li, and Xin Jin. Optogenetic editing reveals the hierarchical organization of learned action sequences. *Cell*, 174(1):32–43, 2018.

Vikash Gilja, Paul Nuyujukian, Cindy A Chestek, John P Cunningham, Byron M Yu, Joline M Fan, Mark M Churchland, Matthew T Kaufman, Jonathan C Kao, Stephen I Ryu, and Krishna V Shenoy. A high-performance neural prosthesis enabled by control algorithm design. *Nature Neuroscience*, 15(12):1752–1757, 2012.

Joshua Glaser, Matthew Whiteway, John P Cunningham, Liam Paninski, and Scott Linderman. Recurrent switching dynamical systems models for multiple interacting neural populations. In *Advances in Neural Information Processing Systems*, volume 33, pages 14867–14878. Curran Associates, Inc., 2020.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

E. Gokcen, J. H. Macke, and B. M. Yu. Disentangling the flow of signals between populations of neurons with delayed latents across groups. *Nature Computational Science*, 2:441–452, 2022.

Evren Gokcen, Anna Ivic Jasper, Alison Xu, Adam Kohn, Christian K. Machens, and Byron M. Yu. Uncovering motifs of concurrent signaling across multiple neuronal populations. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*, pages 24283–24295, 2023. URL https://openreview.net/forum?id=u39QQh5L8Q.

Evren Gokcen, Anna I Jasper, Adam Kohn, Christian K Machens, and Byron M Yu. Fast multi-group gaussian process factor models. *arXiv preprint arXiv:2412.16773*, 2024.

Benjamin A Goldstein, Ann Marie Navar, Michael J Pencina, and John PA Ioannidis. Opportunities and challenges in developing risk prediction models with electronic health records data: a

systematic review. *Journal of the American Medical Informatics Association: JAMIA*, 24(1): 198, 2017.

Rabia Gondur, Usama Bin Sikandar, Evan Schaffer, Mikio Christian Aoi, and Stephen L Keeley. Multi-modal gaussian process variational autoencoders for neural and behavioral data. In *The Twelfth International Conference on Learning Representations*, 2024.

Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, 1993.

Ann M Graybiel. The basal ganglia and chunking of action repertoires. *Neurobiology of learning and memory*, 70(1-2):119–136, 1998.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modelling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Zengcai V Guo, Nuo Li, Daniel Huber, Eran Ophir, Diego Gutnisky, Jonathan T Ting, Guoping Feng, and Karel Svoboda. Flow of cortical activity underlying a tactile decision in mice. *Neuron*, 81(1):179–194, 2014.

Sunil Kumar Gupta, Dinh Phung, Brett Adams, and Svetha Venkatesh. A matrix factorization framework for jointly analyzing multiple nonnegative data sources. In *Data mining for service*, pages 151–170. Springer, 2014.

Kai Han, Yunhe Wang, Chao Zhang, Chao Li, and Chao Xu. Autoencoder inspired unsupervised feature selection. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2941–2945. IEEE, 2018.

Chris Hans. Bayesian Lasso regression. *Biometrika*, 96(4):835–845, 2009.

Harry H Harman. *Modern factor analysis*. University of Chicago press, 1976.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Xiaofei He and Partha Niyogi. Locality preserving projections. In *Advances in neural information processing systems*, pages 153–160, 2004.

Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian score for feature selection. In *Advances in neural information processing systems*, pages 507–514, 2006.

Daniel Hernandez, Antonio Khalil Moretti, Ziqiang Wei, Shreya Saxena, John Cunningham, and Liam Paninski. Nonlinear evolution via spatially-dependent linear dynamics for electrophysiology and calcium data. *arXiv preprint arXiv:1811.02459*, 2018.

Sibylle Hess and Katharina Morik. C-salt: Mining class-specific alterations in boolean matrix factorization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 547–563. Springer, 2017.

Houri Hintiryan, Nicholas N Foster, Ian Bowman, Maxwell Bay, Monica Y Song, Lin Gou, Seita Yamashita, Michael S Bienkowski, Brian Zingg, Muye Zhu, et al. The mouse cortico-striatal projectome. *Nature neuroscience*, 19(8):1100–1114, 2016.

Brian Hopkins and John Gordon Skellam. A new method for determining the type of distribution of plant individuals. *Annals of Botany*, 18(2):213–227, 1954.

H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

Amber Hu, David M. Zoltowski, Aditya Nair, David Anderson, Lea Duncker, and Scott W. Linderman. Modeling latent neural dynamics with gaussian process switching linear dynamical systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://arxiv.org/abs/2408.03330.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

Che-Wei Huang and Shrikanth Narayanan. Comparison of feature-level and kernel-level data fusion methods in multi-sensory fall detection. In *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2016.

Dong Huang, Xiaosha Cai, and Chang-Dong Wang. Unsupervised feature selection with multi-subspace randomization and collaboration. *Knowledge-Based Systems*, 182:104856, 2019.

Barbara J Hunnicutt, Bart C Jongbloets, William T Birdsong, Katrina J Gertz, Haining Zhong, and Tianyi Mao. A comprehensive excitatory input map of the striatum reveals novel functional organization. *elife*, 5:e19103, 2016.

Eun Jung Hwang, Trevor D Link, Yvonne Yuling Hu, Shan Lu, Eric Hou-Jen Wang, Varoth Lilascharoen, Sage Aronson, Keelin O'Neil, Byung Kook Lim, and Takaki Komiyama. Corticostriatal flow of action selection bias. *Neuron*, 104(6):1126–1140, 2019.

Ana R Inácio, Ka Chun Lam, Yuan Zhao, Francisco Pereira, Charles R Gerfen, and Soohyun Lee. Brain-wide presynaptic networks of functionally distinct cortical neurons. *Nature*, pages 1–11, 2025.

Md Rashedul Islam, Aklima Akter Lima, Sujoy Chandra Das, MF Mridha, Akibur Rahman Prodeep, and Yutaka Watanobe. A comprehensive survey on the process, methods, evaluation, and challenges of feature selection. *IEEE Access*, 2022.

Soham Jana, Henry Li, Yutaro Yamada, and Ofir Lindenbaum. Support recovery with projected stochastic gates: Theory and application for linear models. *Signal Processing*, 213:109193, 2023.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkE3y85ee.

Andras Janosi, William Steinbrunn, Matthias Pfisterer, and Robert Detrano. Heart Disease. UCI Machine Learning Repository, 1988. DOI: 10.24432/C52P4X.

Xin Jin, Fatuel Tecuapetla, and Rui M Costa. Basal ganglia subcircuits distinctively encode the parsing and concatenation of action sequences. *Nature neuroscience*, 17(3):423–430, 2014.

Yvonne Johansson and Gilad Silberberg. The functional organization of cortical and thalamic inputs onto five types of striatal neurons is determined by source and target cell identities. *Cell reports*, 30(4):1178–1194, 2020.

Justin Jude, Matthew Perich, Lee Miller, and Matthias Hennig. Robust alignment of cross-session recordings of neural population activity by behaviour via unsupervised domain adaptation. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of the 39th International Conference on Machine Learning*, pages 10462–10475. ICLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/jude22a.html.

J. J. Jun, N. A. Steinmetz, J. H. Siegle, et al. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551:232–236, 2017. doi: 10.1038/nature24636.

Matthew T Kaufman, Mark M Churchland, Stephen I Ryu, and Krishna V Shenoy. Cortical activity in the null space: permitting preparation without movement. *Nature Neuroscience*, 17 (3):440–448, 2014.

Yosi Keller, Ronald R Coifman, Stéphane Lafon, and Steven W Zucker. Audio-visual group recognition using diffusion maps. *IEEE Transactions on Signal Processing*, 58(1):403–413, 2009.

Andrei Khilkevich, Michael Lohse, Ryan Low, Ivana Orsolic, Tadej Bozic, Paige Windmill, and Thomas D Mrsic-Flogel. Brain-wide dynamics linking sensation to action during decision-making. *Nature*, 634(8035):890–900, 2024.

Jason R Klug, Xunyi Yan, Hilary Hoffman, Max D Engelhardt, Fumitaka Osakada, Edward M Callaway, and Xin Jin. Asymmetric cortical projections to striatal direct and indirect pathways distinctly control actions. *Elife*, 12:RP92992, 2025.

Daniel Knowland, Varoth Lilascharoen, Christopher Pham Pacia, Sora Shin, Eric Hou-Jen Wang, and Byung Kook Lim. Distinct ventral pallidal neural populations mediate separate symptoms of depression. *Cell*, 170(2):284–297, 2017.

Ron Kohavi and George H John. Wrappers for feature subset selection. *Artificial intelligence*, 97 (1-2):273–324, 1997.

Takaki Komiyama, Takashi R Sato, Daniel H O'Connor, Ying-Xin Zhang, Daniel Huber, Bryan M Hooks, Mariano Gabitto, and Karel Svoboda. Learning-related fine-scale specificity imaged in motor cortex circuits of behaving mice. *Nature*, 464(7292):1182–1186, 2010.

S. Koukuntla, J. B. Julian, J. C. Kaminsky, et al. Unsupervised discovery of the shared and private geometry in multi-view data. *arXiv preprint*, 2024.

Alexxai V Kravitz, Benjamin S Freeze, Philip RL Parker, Kenneth Kay, Myo T Thwin, Karl Deisseroth, and Anatol C Kreitzer. Regulation of parkinsonian motor behaviours by optogenetic control of basal ganglia circuitry. *Nature*, 466(7306):622–626, 2010.

Geraldine J Kress, Naoki Yamawaki, David L Wokosin, Ian R Wickersham, Gordon MG Shepherd, and D James Surmeier. Convergent cortical innervation of striatal projection neurons. *Nature neuroscience*, 16(6):665–667, 2013.

Vipin Kumar and Sonajharia Minz. Feature selection: A literature review. *Smart Comput. Rev.*, 4:211–229, 2014.

Lukas Landler, Graeme D Ruxton, and E Pascal Malkemper. Advice on comparing two independent samples of circular data in biology. *Scientific reports*, 11(1):1–10, 2021.

Peter Langfelder, Bin Zhang, and Steve Horvath. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics*, 24(5):719–720, 2008.

Antonio H Lara, Gamaleldin F Elsayed, Andrew J Zimnik, John P Cunningham, and Mark M Churchland. Conservation of preparatory neural events in monkey motor cortex regardless of how movement is initiated. *elife*, 7:e31826, 2018.

Beatrice Laurent and Pascal Massart. Adaptive estimation of a quadratic functional by model selection. *Annals of Statistics*, pages 1302–1338, 2000.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Roy R Lederman and Ronen Talmon. Learning the geometry of common latent variables using alternating-diffusion. *Applied and Computational Harmonic Analysis*, 44(3):509–536, 2018.

Eric Kenji Lee, Asım Emre Gül, Greggory Heller, Anna Lakunina, Santiago Jaramillo, Pawel F Przytycki, and Chandramouli Chandrasekaran. Physmap-interpretable in vivo neuronal cell type identification using multi-modal analysis of electrophysiological data. *BioRxiv*, pages 2024–02, 2024.

Jaeeon Lee, Wengang Wang, and Bernardo L Sabatini. Anatomically segregated basal ganglia pathways allow parallel behavioral modulation. *Nature neuroscience*, 23(11):1388–1398, 2020.

M. Lee and V. Pavlovic. Private–shared disentangled multimodal variational autoencoder for learning hybrid latent representations. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021. arXiv:2012.13024.

Stefan M Lemke, Dhakshin S Ramanathan, Ling Guo, Seok Joon Won, and Karunesh Ganguly. Emergent modular neural control drives coordinated motor actions. *Nature neuroscience*, 22 (7):1122–1131, 2019.

Benjamin Lengerich, Caleb N Ellington, Andrea Rubbi, Manolis Kellis, and Eric P Xing. Contextualized machine learning. *arXiv preprint arXiv:2310.11340*, 2023.

Shahar Levy, Maria Lavzin, Hadas Benisty, Amir Ghanayim, Uri Dubin, Shay Achvat, Zohar Brosh, Fadi Aeed, Brett D Mensh, Yitzhak Schiller, et al. Cell-type-specific outcome representation in the primary motor cortex. *Neuron*, 107(5):954–971, 2020.

David D Lewis. Feature selection and feature extraction for text categorization. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*, 1992.

Fan Li, Yiming Yang, and Eric P Xing. From lasso regression to feature vector machine. In *Advances in Neural Information Processing Systems*, pages 779–786, 2006.

Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. Feature selection. *ACM Computing Surveys*, 50(6):1–45, dec 2017. doi: 10.1145/3136625. URL https://doi.org/10.1145%2F3136625.

Jundong Li, Liang Wu, Harsh Dani, and Huan Liu. Unsupervised personalized feature selection. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018a.

Nuo Li, Tsai-Wen Chen, Zengcai V Guo, Charles R Gerfen, and Karel Svoboda. A motor cortex circuit for motor planning and movement. *Nature*, 519(7541):51–56, 2015.

Wei Li, Jianxing Feng, and Tao Jiang. IsoLasso: a LASSO regression approach to RNA-Seq based transcriptome assembly. In *International Conference on Research in Computational Molecular Biology*, pages 168–188. Springer, 2011.

Weihan Li, Chengrui Li, Yule Wang, and Anqi Wu. Multi-region Markovian Gaussian process: An efficient method to discover directional communications across multiple brain regions. In *International Conference on Machine Learning*, pages 28112–28131. PMLR, 2024.

Yang Li and Junier Oliva. Active feature acquisition with generative surrogate models. In *International Conference on Machine Learning*, pages 6450–6459. PMLR, 2021.

Yifeng Li, Chih-Yu Chen, and Wyeth W Wasserman. Deep feature selection: theory and application to identify enhancers and promoters. *Journal of Computational Biology*, 23(5): 322–336, 2016.

Yuan Li, Benjamin Mark, Garvesh Raskutti, and Rebecca Willett. Graph-based regularization for regression problems with highly-correlated designs. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 740–742. IEEE, 2018b.

Zechao Li, Yi Yang, Jing Liu, Xiaofang Zhou, and Hanqing Lu. Unsupervised feature selection using nonnegative spectral analysis. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

Lianjia. Housing price in Beijing, 2017. URL https://www.kaggle.com/datasets/ruiqurm/lianjia.

Ofer Lindenbaum, Moshe Salhov, Arie Yeredor, et al. Gaussian bandwidth selection for manifold learning and classification. *Data Mining and Knowledge Discovery*, 34(6):1676–1712, 2020a. doi: 10.1007/s10618-020-00692-x. URL https://doi.org/10.1007/s10618-020-00692-x.

Ofir Lindenbaum, Arie Yeredor, Moshe Salhov, and Amir Averbuch. Multi-view diffusion maps. *Information Fusion*, 55:127–149, 2020b.

Ofir Lindenbaum, Moshe Salhov, Amir Averbuch, and Yuval Kluger. L0-sparse canonical correlation analysis. In *International Conference on Learning Representations*, 2021a.

Ofir Lindenbaum, Uri Shaham, Erez Peterfreund, Jonathan Svirsky, Nicolas Casey, and Yuval Kluger. Differentiable unsupervised feature selection based on a gated laplacian. *Advances in Neural Information Processing Systems*, 34, 2021b.

Ofir Lindenbaum, Yariv Aizenbud, and Yuval Kluger. Probabilistic robust autoencoders for outlier detection. *The Conference on Uncertainty in Artificial Intelligence (UAI)*, 2024.

Scott W. Linderman, Andrew C. Miller, Ryan P. Adams, David M. Blei, Liam Paninski, and Matthew J. Johnson. Recurrent switching linear dynamical systems, 2016. URL https:

//arxiv.org/abs/1610.08466. arXiv preprint.

Yun Liu, Po-Hsuan Cameron Chen, Jonathan Krause, and Lily Peng. How to read articles that use machine learning: users' guides to the medical literature. *Jama*, 322(18):1806–1816, 2019.

Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=H1Y8hhg0b.

T. A. Machado, I. V. Kauvar, and K. Deisseroth. Multiregion neuronal activity: the forest and the trees. *Nature Reviews Neuroscience*, 23(11):683–704, 2022. doi: 10.1038/s41583-022-00634-0.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=S1jE5L5gl.

Hiroshi Makino, Chi Ren, Haixin Liu, An Na Kim, Neehar Kondapaneni, Xin Liu, Duygu Kuzum, and Takaki Komiyama. Transformation of cortex-wide emergent properties during motor learning. *Neuron*, 94(4):880–890, 2017.

Gil Mandelbaum, Julian Taranda, Trevor M Haynes, Daniel R Hochbaum, Kee Wui Huang, Minsuk Hyun, Kannan Umadevi Venkataraju, Christoph Straub, Wengang Wang, Keiramarie Robertson, et al. Distinct cortical-thalamic-striatal circuits through the parafascicular nucleus. *Neuron*, 102(3):636–652, 2019.

Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *nature*, 503(7474):78–84, 2013.

Principal manuscript editors, Analysis coordination, Integrated data analysis Armand Ethan 42 Yao Zizhen 5, ATAC seq data generation, processing Fang Rongxin 45 Hou Xiaomeng 10 Lucero Jacinta D. 18 Osteen Julia K. 18 Pinto-Duarte Antonio 18 Poirion Olivier 10 Preissl Sebastian 10 Wang Xinxin 10 97 97, Epi retro-seq data generation, processing Dominguez Bertha 53 Ito-Cole Tony 1 Jacobs Matthew 1 Jin Xin 54 99 100 99 100 Lee Cheng-Ta 53 Lee Kuo-Fen 53 Miyazaki Paula Assakura 1 Pang Yan 1 Rashid Mohammad 1 Smith Jared B. 54 Vu Minh 1 Williams Elora 54, et al. A multimodal cell census and atlas of the mammalian primary motor cortex. *Nature*, 598(7879):86–102, 2021.

Jeffrey E Markowitz, Winthrop F Gillis, Celia C Beron, Shay Q Neufeld, Keiramarie Robertson, Neha D Bhagat, Ralph E Peterson, Emalee Peterson, Minsuk Hyun, Scott W Linderman, et al. The striatum organizes 3d behavior via moment-to-moment action selection. *Cell*, 174(1):44–58, 2018.

Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature neuroscience*, 21(9):1281–1289, 2018.

Chengbo Meng, Jingheng Zhou, Amy Papaneri, Teja Peddada, Karen Xu, and Guohong Cui. Spectrally resolved fiber photometry for multi-component analysis of brain circuits. *Neuron*, 98(4):707–717, 2018.

Tong Meng, Xuyang Jing, Zheng Yan, and Witold Pedrycz. A survey on machine learning for data fusion. *Information Fusion*, 57:115–129, 2020.

Andrew Miller, Nick Foti, Alexander D'Amour, and Ryan P Adams. Reducing reparameterization gradient variance. In *Advances in Neural Information Processing Systems*, pages 3708–3718, 2017.

Akinori Mitani and Takaki Komiyama. Real-time processing of two-photon calcium imaging data including lateral motion artifact correction. *Frontiers in neuroinformatics*, 12:98, 2018.

Panagiotis Moutafis, Mengjun Leng, and Ioannis A. Kakadiaris. An overview and empirical comparison of distance metric learning methods. *IEEE Transactions on Cybernetics*, 47(3): 612–625, 2017. doi: 10.1109/TCYB.2016.2521767.

Boaz Nadler, Stephane Lafon, Ioannis Kevrekidis, and Ronald Coifman. Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. *Advances in neural information processing systems*, 18, 2005.

Boaz Nadler, Stéphane Lafon, Ronald R Coifman, and Ioannis G Kevrekidis. Diffusion maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113–127, 2006.

Boaz Nadler, Stephane Lafon, Ronald Coifman, and Ioannis G Kevrekidis. Diffusion maps-a probabilistic interpretation for spectral embedding and clustering algorithms. In *Principal manifolds for data visualization and dimension reduction*, pages 238–260. Springer, 2008.

Anders Nelson, Brenda Abdelmesih, and Rui M Costa. Corticospinal populations broadcast complex motor signals to coordinated spinal and striatal circuits. *Nature neuroscience*, 24(12): 1721–1732, 2021.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

S Ochilov, MS Alam, and A Bal. Fukunaga-koontz transform based dimensionality reduction for hyperspectral imagery. In *Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XII*, volume 6233, pages 829–836. SPIE, 2006.

Wolfgang Omlor, Anna-Sophia Wahl, Pia Sipilä, Henry Lütcke, Balazs Laurenczy, I-Wen Chen, Lazar T Sumanovski, Marcel van't Hoff, Philipp Bethge, Fabian F Voigt, et al. Context-dependent limb movement encoding in neuronal populations of motor cortex. *Nature communications*, 10(1):4812, 2019.

Ofek Ophir, Orit Shefi, and Ofir Lindenbaum. Classifying neuronal cell types based on shared electrophysiological information from humans and mice. *Neuroinformatics*, 22(4):473–486, 2024.

Praisan Padungweang, Chidchanok Lursinsap, and Khamron Sunat. Univariate filter technique for unsupervised feature selection using a new laplacian score based local nearest neighbors. In *2009 Asia-Pacific Conference on Information Processing*, volume 2, pages 196–200. IEEE, 2009.

C. Pandarinath, D. J. O'Shea, J. Collins, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15:805–815, 2018.

Jones G Parker, Jesse D Marshall, Biafra Ahanonu, Yu-Wei Wu, Tony Hyun Kim, Benjamin F Grewe, Yanping Zhang, Jin Zhong Li, Jun B Ding, Michael D Ehlers, et al. Diametric neural ensemble dynamics in parkinsonian and dyskinetic states. *Nature*, 557(7704):177–182, 2018.

Philip RL Parker, Arnaud L Lalive, and Anatol C Kreitzer. Pathway-specific remodeling of thalamostriatal synapses in parkinsonian mice. *Neuron*, 89(4):734–740, 2016.

Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

Matthew G Perich, Juan A Gallego, and Lee E Miller. A neural population mechanism for rapid learning. *Neuron*, 100(4):964–976, 2018.

Matthew G Perich, Charlotte Arlt, Sofia Soares, Megan E Young, Clayton P Mosher, Juri Minxha, Eugene Carter, Ueli Rutishauser, Peter H Rudebeck, Christopher D Harvey, and Kanaka Rajan. Inferring brain-wide interactions using data-constrained recurrent neural network models. *BioRxiv*, pages 2020–12, 2020.

Andrew J Peters, Simon X Chen, and Takaki Komiyama. Emergence of reproducible spatiotemporal activity during motor learning. *Nature*, 510(7504):263–267, 2014.

Andrew J Peters, Haixin Liu, and Takaki Komiyama. Learning in the rodent motor cortex. *Annual review of neuroscience*, 40(1):77–97, 2017.

Andrew J Peters, Julie MJ Fabre, Nicholas A Steinmetz, Kenneth D Harris, and Matteo Carandini. Striatal activity topographically reflects cortical activity. *Nature*, 591(7850):420–425, 2021.

James W Phillips, Anton Schulmann, Erina Hara, Johan Winnubst, Chenghao Liu, Vera Valakh, Lihua Wang, Brenda C Shields, Wyatt Korff, Jayaram Chandrashekar, et al. A repeated molecular architecture across thalamic pathways. *Nature neuroscience*, 22(11):1925–1935, 2019.

Lutz Prechelt. Early stopping—but when? *Neural Networks: Tricks of the Trade*, pages 53–67, 2012.

Yu Qi, Hanwen Wang, Rui Liu, Bian Wu, Yueming Wang, and Gang Pan. Activity-dependent neuron model for noise resistance. *Neurocomputing*, 357:240–247, 2019.

Mingjie Qian and Chengxiang Zhai. Robust unsupervised feature selection. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

V Madhusudan Rao and VN Sastry. Unsupervised feature ranking based on representation entropy. In *2012 1st International Conference on Recent Advances in Information Technology (RAIT)*, pages 421–425. IEEE, 2012.

Peter Redgrave, Tony J Prescott, and Kevin Gurney. The basal ganglia: a vertebrate solution to the selection problem? *Neuroscience*, 89(4):1009–1023, 1999.

Chi Ren, Kailong Peng, Ruize Yang, Weikang Liu, Chang Liu, and Takaki Komiyama. Global and subtype-specific modulation of cortical inhibitory neurons regulated by acetylcholine during motor learning. *Neuron*, 110(14):2334–2350, 2022.

Shanna L Resendez, Josh H Jennings, Randall L Ung, Vijay Mohan K Namboodiri, Zhe Charles Zhou, James M Otis, Hiroshi Nomura, Jenna A McHenry, Oksana Kosyk, and Garret D Stuber. Visualization of cortical, subcortical and deep brain neural circuit dynamics during naturalistic mammalian behavior with head-mounted microscopes and chronically implanted lenses. *Nature protocols*, 11(3):566–597, 2016.

Juha Reunanen. Overfitting in making comparisons between variable selection methods. *Journal of Machine Learning Research*, 3(Mar):1371–1382, 2003.

Mattia Rigotti, Omri Barak, Melissa R Warden, Xiao-Jing Wang, Nathaniel D Daw, Earl K Miller, and Stefano Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585–590, 2013.

Omid G Sani, Hamidreza Abbaspourazad, Yan T Wong, Bijan Pesaran, and Maryam M Shanechi. Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. *Nature neuroscience*, 24(1):140–149, 2021.

Arpiar Saunders, Evan Z Macosko, Alec Wysoker, Melissa Goldman, Fenna M Krienen, Heather de Rivera, Elizabeth Bien, Matthew Baum, Laura Bortolin, Shuyu Wang, et al. Molecular diversity and specializations among the cells of the adult mouse brain. *Cell*, 174(4):1015–1030, 2018.

Shreya Saxena and John P Cunningham. Towards the neural population doctrine. *Current opinion in neurobiology*, 55:103–111, 2019.

Simone Scardapane, Danilo Comminiello, Amir Hussain, and Aurelio Uncini. Group sparse regularization for deep neural networks. *Neurocomput.*, 241(C):81–89, June 2017. ISSN 0925-2312. doi: 10.1016/j.neucom.2017.02.029. URL https://doi.org/10.1016/j.neucom.2017.02.029.

Aidan Schneider, Mehdi Azabou, Louis McDougall-Vigier, David F Parks, Sahara Ensley, Kiran Bhaskaran-Nair, Tomasz Nowakowski, Eva L Dyer, and Keith B Hengen. Transcriptomic cell type structures in vivo neuronal activity across multiple timescales. *Cell reports*, 42(4), 2023a.

Steffen Schneider, Jin Hwa Lee, and Mackenzie Weygandt Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, 2023b.

João D Semedo, Amir Zandvakili, Christian K Machens, Byron M Yu, and Adam Kohn. Cortical areas interact through a communication subspace. *Neuron*, 102(1):249–259.e4, 2019.

Uri Shaham, Kelly Stanton, Henry Li, Ronen Basri, Boaz Nadler, and Yuval Kluger. Spectralnet: Spectral clustering using deep neural networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HJ_aoCyRZ.

Uri Shaham, Ofir Lindenbaum, Jonathan Svirsky, and Yuval Kluger. Deep unsupervised feature selection by discarding nuisance and correlated features. *Neural Networks*, 152:34–43, 2022.

Alok Sharma, Kuldip K Paliwal, Seiya Imoto, and Satoru Miyano. A feature selection method using improved regularized linear discriminant analysis. *Machine vision and applications*, 25 (3):775–786, 2014.

Yiyuan She. *Sparse regression with exact clustering*. Stanford University, 2008.

Xilin Shen, Fuyuze Tokoglu, Xenios Papademetris, and R Todd Constable. Groupwise whole-brain parcellation from resting-state fMRI data for network node identification. *Neuroimage*, 82:403–415, 2013.

Meng-jun Sheng, Di Lu, Zhi-ming Shen, and Mu-ming Poo. Emergence of stable striatal d1r and d2r neuronal ensembles with distinct firing sequence during motor learning. *Proceedings of the National Academy of Sciences*, 116(22):11038–11047, 2019.

Yuge Shi, Brooks Paige, Philip Torr, et al. Variational mixture-of-experts autoencoders for multi-modal deep generative models. *Advances in neural information processing systems*, 32, 2019.

Hajin Shim, Sung Ju Hwang, and Eunho Yang. Joint active feature acquisition and classification with variable-size set encoding. *Advances in neural information processing systems*, 31, 2018.

Tal Shnitzer, Mirela Ben-Chen, Leonidas Guibas, Ronen Talmon, and Hau-Tieng Wu. Recovering hidden components in multimodal data with composite diffusion operators. *SIAM Journal on Mathematics of Data Science*, 1(3):588–616, 2019.

Leslie J Sibener, Alice C Mosberger, Tiffany X Chen, Vivek R Athalye, James M Murray, and Rui M Costa. Dissociable roles of distinct thalamic circuits in learning reaches to spatial targets. *Nature Communications*, 16(1):2962, 2025.

Kimberly Siletti, Rebecca Hodge, Alejandro Mossi Albiach, Ka Wai Lee, Song-Lin Ding, Lijuan Hu, Peter Lönnerberg, Trygve Bakken, Tamara Casper, Michael Clark, et al. Transcriptomic diversity of cell types across the adult human brain. *Science*, 382(6667):eadd7046, 2023.

Amit Singer, Radek Erban, Ioannis G Kevrekidis, and Ronald R Coifman. Detecting intrinsic slow variables in stochastic dynamical systems by anisotropic diffusion maps. *Proceedings of the National Academy of Sciences*, 106(38):16090–16095, 2009.

Deepak Singla, Long Yang, Andrew S Weakley, Dylan Davidoff, Jonathan C Kao, and Sotiris C Masmanidis. Complementary cortical and striatal encoding of locomotor preparation and performance. *bioRxiv*, pages 2025–04, 2025.

Fengxi Song, Dayong Mei, and Hongfeng Li. Feature selection based on linear discriminant analysis. In *2010 international conference on intelligent system design and engineering application*, volume 1, pages 746–749. IEEE, 2010.

Le Song, Alex Smola, Arthur Gretton, Karsten M Borgwardt, and Justin Bedo. Supervised feature selection via dependence estimation. In *Proceedings of the 24th international conference on Machine learning*, pages 823–830. ACM, 2007.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(May):1393–1434, 2012.

Ram Dyuthi Sristi, Gal Mishne, and Ariel Jaffe. Disc: Differential spectral clustering of features. In *Advances in Neural Information Processing Systems*, 2022.

Barry E. Stein and M. Alex Meredith. *The Merging of the Senses*. MIT Press, Cambridge, MA, 1993.

Gary Stein, Bing Chen, Annie S Wu, and Kien A Hua. Decision tree classifier for network intrusion detection with ga-based feature selection. In *Proceedings of the 43rd annual Southeast regional conference-Volume 2*, pages 136–141. ACM, 2005.

Nicholas A Steinmetz, Peter Zatka-Haas, Matteo Carandini, and Kenneth D Harris. Distributed coding of choice, action and engagement across the mouse brain. *Nature*, 576(7786):266–273, 2019.

Mark G Stokes, Makoto Kusunoki, Natasha Sigala, Hamed Nili, David Gaffan, and John Duncan. Dynamic coding for cognitive control in prefrontal cortex. *Neuron*, 78(2):364–375, 2013.

Thomas M Sutter, Imant Daunhawer, and Julia E Vogt. Generalized multimodal elbo. *International Conference on Learning Representations*, 2021.

Ronen Talmon and Hau-Tieng Wu. Latent common manifold learning with alternating diffusion: analysis and applications. *Applied and Computational Harmonic Analysis*, 47(3):848–892, 2019.

Fatuel Tecuapetla, Xin Jin, Susana Q Lima, and Rui M Costa. Complementary contributions of striatal projection pathways to action initiation and execution. *Cell*, 166(3):703–715, 2016.

D Gowanlock R Tervo, Elena Kuleshova, Maxim Manakov, Mikhail Proskurin, Mattias Karlsson, Andy Lustig, Reza Behnam, and Alla Y Karpova. The anterior cingulate cortex directs exploration of alternative strategies. *Neuron*, 109(11):1876–1887, 2021.

Ryan Thompson, Amir Dezfouli, and Robert Kohn. The contextual lasso: Sparse linear models via deep neural networks, 2023.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996a.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996b.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Parsa Vahidi, Omid G Sani, and Maryam Shanechi. Braid: Input-driven nonlinear dynamical modeling of neural-behavioral data. In *The Thirteenth International Conference on Learning Representations*, 2024.

Parsa Vahidi, Omid G Sani, and Maryam M Shanechi. Braid: Input-driven nonlinear dynamical modeling of neural-behavioral data. *arXiv preprint arXiv:2509.18627*, 2025.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008.

David C Van Essen, Stephen M Smith, Deanna M Barch, Timothy EJ Behrens, Essa Yacoub, Kamil Ugurbil, Wu-Minn HCP Consortium, et al. The WU-Minn human connectome project: an overview. *Neuroimage*, 80:62–79, 2013.

Roy Varshavsky, Assaf Gottlieb, Michal Linial, and David Horn. Novel unsupervised feature filtering of biological data. *Bioinformatics*, 22(14):e507–e513, 2006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Saurabh Vyas, Matthew D Golub, David Sussillo, and Krishna V Shenoy. Computation through neural population dynamics. *Annual review of neuroscience*, 43(1):249–275, 2020.

Nicholas R Wall, Mauricio De La Parra, Edward M Callaway, and Anatol C Kreitzer. Differential innervation of direct-and indirect-pathway striatal projection neurons. *Neuron*, 79(2):347–360, 2013.

Qian Wang and Toby P Breckon. Generalized zero-shot domain adaptation via coupled conditional variational autoencoders. *Neural Networks*, 163:40–52, 2023.

Shuyang Wang, Zhengming Ding, and Yun Fu. Feature selection guided auto-encoder. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Suhang Wang, Jiliang Tang, and Huan Liu. Embedded unsupervised feature selection. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

W. Wang, R. Arora, K. Livescu, and J. Bilmes. On deep multi-view representation learning: Objectives and optimization. In *Proc. of the 33rd International Conference on Machine Learning*, pages 1083–1092, 2016.

Wen Wang, Zhen Cui, Hong Chang, Shiguang Shan, and Xilin Chen. Deeply coupled auto-encoder networks for cross-view classification. *arXiv preprint arXiv:1402.2031*, 2014.

Joe H Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American*

statistical association, 58(301):236–244, 1963.

Geoffrey S Watson. Goodness-of-fit tests on a circle. ii. *Biometrika*, 49(1/2):57–63, 1962.

Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016.

Ian R Wickersham, Stefan Finke, Karl-Klaus Conzelmann, and Edward M Callaway. Retrograde neuronal tracing with a deletion-mutant rabies virus. *Nature methods*, 4(1):47–49, 2007a.

Ian R Wickersham, David C Lyon, Richard JO Barnard, Takuma Mori, Stefan Finke, Karl-Klaus Conzelmann, John AT Young, and Edward M Callaway. Monosynaptic restriction of transsynaptic tracing from single, genetically targeted neurons. *Neuron*, 53(5):639–647, 2007b.

Daniela M Witten, Ali Shojaie, and Fan Zhang. The cluster elastic net for high-dimensional regression with unknown variable grouping. *Technometrics*, 56(1):112–122, 2014.

Lin Wu, Yang Wang, and Ling Shao. Cycle-consistent deep generative hashing for cross-modal retrieval. *IEEE Transactions on Image Processing*, 28(4):1602–1612, 2018.

Mike Wu and Noah Goodman. Multimodal generative models for scalable weakly-supervised learning. *Advances in neural information processing systems*, 31, 2018.

Li Xiao, Julia M Stephen, Tony W Wilson, Vince D Calhoun, and Yu-Ping Wang. Alternating diffusion map based fusion of multimodal brain connectivity networks for IQ prediction. *IEEE Transactions on Biomedical Engineering*, 66(8):2140–2151, 2018.

Yutaro Yamada, Ofir Lindenbaum, Sahand Negahban, and Yuval Kluger. Feature selection using stochastic gates. In *International Conference on Machine Learning*, pages 10648–10659. PMLR, 2020.

Junchen Yang, Ofir Lindenbaum, and Yuval Kluger. Locally sparse neural networks for tabular biomedical data. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 25123–25153. PMLR, 17–23 Jul 2022a. URL https://proceedings.mlr.press/v162/yang22i.html.

Junchen Yang, Ofir Lindenbaum, and Yuval Kluger. Locally sparse neural networks for tabular biomedical data. In *International Conference on Machine Learning*, pages 25123–25153. PMLR, 2022b.

Long Yang, Deepak Singla, Alexander K Wu, Katy A Cross, and Sotiris C Masmanidis.

Dopamine lesions alter the striatal encoding of single-limb gait. *Elife*, 12:RP92821, 2024.

Weiguo Yang, Sri Laasya Tipparaju, Guang Chen, and Nuo Li. Thalamus-driven functional populations in frontal cortex support decision-making. *Nature neuroscience*, 25(10):1339–1352, 2022c.

Zizhen Yao, Cindy TJ Van Velthoven, Michael Kunst, Meng Zhang, Delissa McMillen, Changkyu Lee, Won Jung, Jeff Goldy, Aliya Abdelhak, Matthew Aitken, et al. A high-resolution transcriptomic and spatial atlas of cell types in the whole mouse brain. *Nature*, 624(7991): 317–332, 2023.

Joel Ye and Chethan Pandarinath. Representation learning for neural population activity with neural data transformers, 2021. URL https://arxiv.org/abs/2108.01210. arXiv preprint.

Daiyao Yi, Hao Dong, Michael James Higley, Anne Churchland, and Shreya Saxena. Shared-ae: Automatic identification of shared subspaces in high-dimensional neural and behavioral activity. In *The Thirteenth International Conference on Learning Representations*, 2025.

Henry H Yin and Barbara J Knowlton. The role of the basal ganglia in habit formation. *Nature reviews neuroscience*, 7(6):464–476, 2006.

Mingzhang Yin, Nhat Ho, Bowei Yan, Xiaoning Qian, and Mingyuan Zhou. Probabilistic best subset selection via gradient-based optimization, 2022.

Jinsung Yoon, James Jordon, and Mihaela van der Schaar. Invase: Instance-wise variable selection using neural networks. In *International Conference on Learning Representations*, 2018.

Eric A Yttri and Joshua T Dudman. Opponent and bidirectional control of movement velocity in the basal ganglia. *Nature*, 533(7603):402–406, 2016.

B. M. Yu, J. P. Cunningham, G. Santhanam, et al. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Advances in Neural Information Processing Systems*, pages 1881–1888, 2009.

Yi Yu, Tengyao Wang, and Richard J Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

Amin Zandvakili and Adam Kohn. Coordinated neuronal activity enhances corticocortical communication. *Neuron*, 87(4):827–839, 2015.

Luke Zappia, Belinda Phipson, and Alicia Oshlack. Splatter: simulation of single-cell rna sequencing data. *Genome biology*, 18(1):1–15, 2017.

Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. *Advances in neural information processing systems*, 17, 2004.

Lihi Zelnik-Manor and Pietro Perona. Self-tuning spectral clustering. In *Advances in neural information processing systems*, pages 1601–1608, 2005.

Hong Zeng and Yiu-ming Cheung. Feature selection and kernel learning for local learning-based clustering. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1532–1547, 2010.

Baibing Zhang, Claire E Geddes, and Xin Jin. Complementary corticostriatal circuits orchestrate action repetition and switching. *Science Advances*, 11(21):eadt0854, 2025.

Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1058–1067, 2017.

Jun Zhao, Ariel Jaffe, Henry Li, Ofir Lindenbaum, Esen Sefik, Ruaidhrí Jackson, Xiuyuan Cheng, Richard A Flavell, and Yuval Kluger. Detection of differentially abundant cell subpopulations in scrna-seq data. *Proceedings of the National Academy of Sciences*, 118(22), 2021.

Zheng Zhao and Huan Liu. Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th international conference on Machine learning*, pages 1151–1157, 2007.

Zheng Alan Zhao and Huan Liu. *Spectral Feature Selection for Data Mining (Open Access)*. CRC Press, 2011.

Zexuan Zhu, Yew-Soon Ong, and Manoranjan Dash. Wrapper–filter feature selection algorithm using a memetic framework. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(1):70–76, 2007.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.