

Deep Clustering of Animal Motion Tracking Data:
software development and applications to
psychiatric preclinical research

Lucas Miranda

Vollständiger Abdruck der von der TUM School of Medicine and Health der Technischen Universität München zur Erlangung eines

Doktors der Naturwissenschaften
(Dr. rer. nat.)

Vorsitz:

Prüfende der Dissertation:

1. Prof. Dr. Bertram Müller-Myhsok
2. Prof. Dr. Julien Gagneur

Die Dissertation wurde am 18.07.2023 bei der Technischen Universität München eingereicht und durch die TUM School of Medicine and Health am 08.11.2023 angenommen.

Acknowledgements

Acknowledgements

*To Tomás
To Sol
To Pedro
To Tamayo, Stella, Olga, and Tito
To Flor
To Fabio and Silvina
who told me the sky was the limit
and taught me how to fly*



The work presented here would not have been possible without the help and support of Benno Pütz (my great mentor and travel companion), Joeri Bordes (whose hard work and contributions were invaluable in conceiving and implementing DeepOF and its applications), Mathias V. Schmidt, and, of course, Bertram, who let me make my own path and explore interdisciplinary collaborations, while always being there for me.

Along these lines, I feel the need to highlight the beautiful collaborative atmosphere within the Max Planck Institute of Psychiatry and the International Max Planck Research School for Translational Psychiatry (IMPRS-TP). Most (if not all) of my work was sparked by interactions with other research groups within these institutions, where complementary expertise and the learning of a common language were (and continue to be) key to reaching our humble goals. Special thanks to Darina Czamara and Juan Pablo Lopez, who supported me with advice and presented me with opportunities at all times, and to Elisabeth Binder.

On top of this, I was fortunate enough to have been part of the Machine Learning Frontiers in Precision Medicine Marie Curie Innovative Training Network (MLFPM-ITN). Here, people like Felix Agakov, Karsten Borgwardt, Katharina Heinrich, and many others, made sure I and all students had everything we needed to thrive in our own individual and collective scientific endeavors. For this I am immensely grateful.

Last but not least, I thank my friends and family back home. Their unconditional support, love, and company in the distance were key to finding motivation, especially through the difficult times many of us endured in isolation over the last couple of years.

Abstract

Abstract

Attempts to systematically characterize and understand how living organisms react to complex stimuli are not new. Until recently, however, most approaches have either relied on observational studies, which prevent researchers from testing specific hypotheses, or in overly simplified and laborious laboratory settings, that are too far from real-world scenarios. Along these lines, a recent trend has been to recreate semi-naturalistic scenarios in a controlled manner, and use new technologies to extract information from freely moving animals such as motion tracking, neural activity, and vocalization, among others.

In the field of motion tracking, and leveraging advances in machine learning, particularly in neural networks used for computer vision, several openly available software packages have recently started to provide tools that require little effort to accurately track multiple body parts over time, without the need for physical markers. This has opened the way for researchers to obtain large amounts of data with little effort, which has in turn helped developers come up with novel ways to analyze and extract insight from this novel data source.

Thus and so, the current thesis aims to provide three main contributions. First, to develop novel deep clustering algorithms specifically tailored to this type of time series, that can be used to explore the behavioral repertoire of animals without the need of human labels. Second, the deployment of these algorithms in an open-source Python package, which includes them alongside other tools to annotate the behavior of laboratory rodents. Third and last, to use the deployed algorithms to characterize a real world animal model.

Moreover, this is a publication-based dissertation, which presents the accomplished results as two papers accepted for publication in peer-reviewed journals. The first two mentioned goals are addressed in an article published in the Journal of Open Source Software (JOSS), titled “**DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data**”. Here, we present an originally developed software tool called DeepOF (Deep Open Field), which includes several deep clustering algorithms alongside other tools, and is ready for researchers to use.

The second paper was published in Nature Communications, and is titled “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”. Here, we present a characterization of Chronic Social Defeat Stress (CSDS), an animal model widely used in stress and depression research, using the novel software presented in the first article.

All in all, this thesis provides a set of novel contributions to both the behavioral research field in general, and the analysis of motion tracking data in particular. The next few chapters will describe these contributions in detail, and how I believe they hold the potential to positively impact future research.

Zusammenfassung

Zusammenfassung

Versuche, systematisch zu charakterisieren und zu verstehen, wie lebende Organismen auf komplexe Reize reagieren, sind nicht neu. Bis vor kurzem stützten sich die meisten Ansätze jedoch entweder auf Beobachtungsstudien, die es den Forschern unmöglich machen, spezifische Hypothesen zu testen, oder auf allzu vereinfachte und mühsame Laborsituationen, die zu weit von realen Szenarien entfernt sind. In diesem Sinne geht der Trend in letzter Zeit dahin, halbnatürliche Szenarien kontrolliert nachzustellen und neue Technologien einzusetzen, um Informationen von sich frei bewegenden Tieren zu extrahieren, wie z.B. Bewegungsverfolgung, neuronale Aktivität und Vokalisierung.

Auf dem Gebiet der Bewegungsverfolgung und unter Ausnutzung der Fortschritte im Bereich des maschinellen Lernens, insbesondere bei neuronalen Netzen, die für das Computersehen verwendet werden, haben mehrere frei verfügbare Softwarepakete in letzter Zeit begonnen, Tools bereitzustellen, die mit geringem Aufwand eine genaue Verfolgung mehrerer Körperteile über die Zeit ermöglichen, ohne dass physische Marker erforderlich sind. Dies hat Forschern die Möglichkeit eröffnet, mit geringem Aufwand große Datenmengen zu erhalten, was wiederum Entwicklern geholfen hat, neue Wege zu finden, um diese neuartige Datenquelle zu analysieren und Erkenntnisse daraus zu gewinnen.

Die vorliegende Arbeit zielt also darauf ab, drei Hauptbeiträge zu leisten. Erstens die Entwicklung neuartiger Deep Clustering-Algorithmen, die speziell auf diese Art von Zeitreihen zugeschnitten sind und mit denen sich das Verhaltensrepertoire von Tieren ohne menschliche Kennzeichnung erforschen lässt. Zweitens, die Bereitstellung dieser Algorithmen in einem Open-Source-Python-Paket, das sie zusammen mit anderen Tools zur Annotation des Verhaltens von Labornagern enthält. Drittens und letztens, die Verwendung der eingesetzten Algorithmen zur Charakterisierung eines realen Tiermodells.

Darüber hinaus handelt es sich um eine publikationsbasierte Dissertation, in der die erzielten Ergebnisse in Form von zwei zur Veröffentlichung in begutachteten Fachzeitschriften angenommenen Artikeln präsentiert werden. Die ersten beiden genannten Ziele werden in einem im Journal of Open Source Software (JOSS) veröffentlichten Artikel mit dem Titel **“DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data”** behandelt. Hier stellen wir ein ursprünglich entwickeltes Software-Tool namens DeepOF (Deep Open Field) vor, das neben anderen Tools auch mehrere Deep-Clustering-Algorithmen enthält und für Forscher einsatzbereit ist.

Die zweite Arbeit wurde in Nature Communications veröffentlicht und trägt den Titel **“Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress”**. Hier stellen wir eine Charakterisierung von Chronic Social Defeat Stress (CSDS) vor, einem in der Stress- und Depressionsforschung weit verbreiteten Tiermodell, bei dem die im ersten Artikel vorgestellte neue Software zum Einsatz kommt.

Alles in allem liefert diese Arbeit eine Reihe neuartiger Beiträge sowohl zur Verhaltensforschung im Allgemeinen als auch zur Analyse von Motion-Tracking-Daten im Besonderen. In den nächsten Kapiteln werden diese Beiträge im Detail beschrieben und wie ich glaube, dass sie das Potenzial haben, die zukünftige Forschung positiv zu beeinflussen.

Contents

Acknowledgements	iii
Abstract	vii
Zusammenfassung	ix
Contents	xi
List of Figures	xv
List of Tables	xix
Acronyms	xxi
1 Introduction	1
1.1 Clinical and preclinical research: altered behavior in psychiatric conditions	2
1.1.1 Defining behavior	2
1.1.2 A brief history of psychiatry	3
1.1.3 Bridging the translational gap: from animal models back to humans	4
1.2 Quantifying behavior	5
1.2.1 Endless forms most beautiful: from ethology to controlled behavioral experiments	6
1.2.2 Deep learning and the advent of markerless pose estimation	8
1.3 Automated annotation of motion tracking data	11
1.3.1 Supervised annotation	11
1.3.2 Unsupervised annotation and behavioral embeddings	13
1.4 Chronic stress as a case study	16
1.4.1 A primer on stress biology	16
1.4.2 Chronic Social Defeat Stress (CSDS)	17
2 State of the Art	21
2.1 Time series clustering	22
2.1.1 Classical methods	24
2.1.1.1 Classical clustering using time-aware feature extraction	24
2.1.1.2 Dynamic Time Warping and temporal K-Means	25
2.1.1.3 Hidden Markov Models	26

CONTENTS

2.1.2	Deep clustering	29
2.1.2.1	Recurrent Neural Networks	30
2.1.2.2	Temporal Convolutional Networks	32
2.1.2.3	Transformer Networks	34
2.2	Segmenting behavior: exploring available approaches	35
2.2.0.1	B-SOiD: time series clustering using guided representations	35
2.2.0.2	MoSeq: motion clustering with Autoregressive Hidden Markov Models	37
2.2.0.3	VAME: Variational Animal Motion Embeddings	38
2.3	Main contributions of this thesis to the field	40
2.3.1	Implementation and testing of novel deep clustering algorithms for unsupervised behavioral segmentation	40
2.3.2	Deployment of the developed algorithms to the community	40
2.3.3	Application of the developed algorithms to the characterization of Chronic Social Defeat Stress	40
3	Methods	41
3.1	Software architecture and deployment	42
3.2	Data loading and input	42
3.3	Time series processing	43
3.4	Supervised annotation of pre-defined traits	43
3.5	Unsupervised annotation: exploring the behavioral space	43
3.5.1	Matrix input/output representations	43
3.5.2	Graph input/output representations	43
3.6	Unsupervised annotation: deep clustering models	44
3.6.1	Variational Deep Embeddings (VaDE)	45
3.6.2	Vector Quantization Variational Autoencoders (VQVAE)	47
3.6.3	Contrastive representation learning (CRL)	48
3.6.4	Semi-supervised post-hoc reclustering	49
3.7	Characterization of Chronic Social Defeat Stress (CSDS)	50
3.8	DeepOF in practice and post-hoc analysis of annotation results	50
3.9	Statistics	50
4	DeepOF: a Python package for pattern recognition in mice motion tracking data	51
4.1	Overview	52
4.2	Package design	52
4.3	Contribution to the work	52
5	Characterizing CSDS using automatically annotated motion tracking data	57
5.1	Overview	58
5.2	Contribution to the work	58

6 Discussion	101
6.1 There and back again: towards systematic quantification of natural behavior	102
6.2 DeepOF in context	103
6.3 Perspectives on supervised learning on behavioral data	104
6.4 Perspectives on unsupervised learning on behavioral data	105
6.5 Increasing resolution in neurobiological research: behavioral quantification in context	106
6.6 Beyond motion tracking: integrating multimodal data	107
6.7 Impact of the presented results in chronic stress research	108
6.8 Frontiers of the field: between translational research and knowledge discovery	110
7 Conclusion & Outlook	113
Bibliography	115
List of published PhD papers	133

List of Figures

1.1	Depression-like syndrome in mice	5
1.2	From ethology to modern behavioral quantification	8
1.3	Common univariate laboratory behavioral tasks	9
1.4	DeepLabCut overview: novel algorithms for markerless motion tracking	12
1.5	Automatic behavioral annotation via motion tracking: exploring different annotation options	15
1.6	From macro to micro: increasing resolution in stress neurobiology	18
1.7	A paradigm shift in translational psychiatry through rodent neuroethology	19
2.1	Issues with time series clustering in Euclidean space	23
2.2	An overview of the tsfresh feature extraction pipeline	25
2.3	Comparison between DTW and Euclidean distance	27
2.4	Hidden Markov Models (HMMs)	28
2.5	Recurrent Neural Networks (RNNs)	32
2.6	Architectural elements of a Temporal Convolutional Network (TCN)	34
2.7	Details on the transformer architecture	36
2.8	Overview of the B-SOiD pipeline	37
2.9	MoSeq: motion clustering with Autorregressive Hidden Markov Models	38
2.10	VAME: Variational Animal Motion Embeddings	39
3.1	Time series input representation for deep clustering of motion tracking data	45
3.2	Deep clustering architectures implemented within DeepOF	46
4.1	Schematic representation of the DeepOF workflow	54
5.1	DeepOF workflow in detail: an overview of data preprocessing and annotation pipelines	61
5.2	Classical hallmarks for chronic social defeat stress	62
5.3	Social interaction binning yields more separable PCA projections than the social avoidance task	63

LIST OF FIGURES

5.4	Top contributing behaviors in the social interaction task for 10 min total duration and time bins	64
5.5	Z-score correlation analysis and the exploration of susceptibility and resiliency	65
5.6	Single-animal unsupervised analyses identify different behavioral patterns between stressed and non-stressed mice during the SI task	67
5.7	SHAP analysis of unsupervised cluster assignments in the single-animal social interaction task	68
5.8	Validation of rule-based annotated behaviors in DeepOF	78
5.9	Validation of the “stopped and huddled” classifier provided within DeepOF	79
5.10	DeepOF behavioral classifiers in the open field task	81
5.11	DeepOF other behavioral classifiers in the social interaction task for 10 min duration	83
5.12	Multi-animal unsupervised analyses identify different two-mice behavioral patterns between arenas containing stressed and non-stressed mice during the SI task	84
5.13	Single-animal unsupervised analyses identify different behavioral patterns between stressed and non-stressed mice during the OF task	86
5.14	Single-animal unsupervised analyses identify mild behavioral differences between stressed and non-stressed mice during the SA task	88
5.15	Global single-animal embeddings across non-overlapping time bins in the SI dataset	89
5.16	Global multi-animal embeddings across non-overlapping time bins in the SI dataset	90
5.17	Cluster enrichment per experimental condition in the second to fourth optimal bins for the single-animal embeddings on the SI task	91
5.18	Cluster enrichment per experimental condition in the second to fourth optimal bins reported for the multi-animal embeddings on the SI task	92
5.19	Spatial distribution of clusters obtained using single-animal embeddings in the SI task	93
5.20	Spatial distribution of clusters obtained using multi-animal embeddings in the SI task	94
5.21	Spatial distribution of clusters obtained in the OF task	95
5.22	Correlation between behavioral entropy and stress physiology Z-score	96
5.23	SHAP analysis of unsupervised cluster assignments in the multi-animal social interaction task	97

5.24 **SHAP analysis of unsupervised cluster assignments in the open field task** 99

List of Tables

5.1	Default thresholds used by the annotation pipeline in DeepOF	80
5.2	Datasets used in the CSDS characterization study	80

Acronyms

AI	Artificial Intelligence.
ANN	Artificial Neural Network.
AR-HMM	Autoregressive Hidden Markov Model.
CI	Continuous Integration.
CRL	Contrastive Representation Learning.
CSDS	Chronic Social Defeat Stress.
DBSCAN	Density-based spatial clustering of applications with noise.
DeepOF	Deep Open Field.
DLC	DeepLabCut.
DLS	Depressive-like Syndrome.
DNN	Deep Neural Network.
DTW	Dynamic Time Warping.
ELBO	Evidence Lower Bound.
fMRI	functional Magnetic Resonance Imaging.
FPS	Frames Per Second.
GAN	Generative Adversarial Network.
GBM	Gradient Boosting Machine.
GMM	Gaussian Mixture Model.
GNN	Graph Neural Network.
HDBSCAN	Hierarchical Density-based spatial clustering of applications with noise.
HMM	Hidden Markov Model.
HPA axis	Hypothalamic-Pituitary-Adrenal axis.
ICD	International Classification of Diseases.
JOSS	Journal of Open Source Software.
KLD	Kullback-Leibler Divergence.

Acronyms

KNN	K-Nearest Neighbors.
LLM	Large Language Model.
MDD	Major Depressive Disorder.
ML	Machine Learning.
MRI	Magnetic Resonance Imaging.
NCE	Noise Contrastive Estimation.
OF	Open Field.
PCA	Principal Component Analysis.
PTSD	Post-Traumatic Stress Disorder.
QTL	Quantitative Trait Loci.
R-CNN	Region-based Convolutional Neural Network.
RDoC	Research Domain Criteria.
RFID	Radio-frequency identifier.
RNN	Recurrent Neural Network.
SA	Social Avoidance.
SHAP	SHapley Additive exPlanations.
SI	Social Interaction.
SLEAP	Social LEAP Estimates Animal Poses.
SNS	Sympathetic Nervous System.
ST-GNN	Spatio-Temporal Graph Neural Network.
TCN	Temporal Convolutional Network.
tsfresh	Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests.
UMAP	Uniform Manifold Approximation and Projection.
VaDE	Variational Deep Embeddings.
VAE	Variational AutoEncoder.
VAME	Variational Animal Motion Embeddings.
VQVAE	Vector Quantization Variational AutoEncoder.
VR	Virtual Reality.

1 Introduction

1.1 Clinical and preclinical research: altered behavior in psychiatric conditions

1.1.1 Defining behavior

Behavior is a fascinating and complex phenomenon which lies at the core of our existence as living organisms. The term refers to the manifestation of an individual's actions and reactions in response to external or internal stimuli, ultimately shaping interactions with the environment and other living things. Our understanding of behavior has evolved over time, and from a biological perspective, it encompasses an intricate interplay between genetic, neurobiological, and physiological processes [1].

To fully grasp the concept of behavior, it is essential to recognize the diverse nature of biological systems involved in its regulation. At the genetic level, heredity and individual variations in genetic makeup play a crucial role in shaping behavioral traits [2]. Genes can influence behavior through the expression of specific proteins, which in turn participate in the development, structure, and function of the nervous system [3]. While it is important to consider the impact of genetic factors, it is also necessary to acknowledge the interaction between genetics and environmental influences on behavior. This interplay, referred to as gene-environment interaction, highlights the dynamic nature of behavior and the continuous adaptation of living organisms to their surroundings [4].

Moreover, a central component of behavior is the nervous system, responsible for receiving, processing, and transmitting information. It constitutes a highly organized network of specialized cells, such as neurons, which communicate with one another through complex electrochemical signaling, allowing for the integration and processing of sensory inputs, generation of responses, and modulation of behavior. Along these lines, neurotransmitters, the chemical messengers facilitating communication between neurons, also play a vital role in the regulation of behavior. These molecules, released by neurons, bind to specific receptors on the receiving cell, initiating a cascade of events that may either excite or inhibit the cell [5].

Furthermore, another critical aspect of behavior is the interplay between the nervous and the endocrine systems. The latter is responsible for the production and release of hormones: chemical messengers secreted by endocrine glands that travel through the bloodstream and exert their effects on target cells [6]. Hormones can influence behavior by acting on the brain and other tissues, modulating emotions, mood, and stress responses [7]. Examples of hormones with significant impact on behavior include cortisol, which is involved in the stress response [8], and oxytocin, which plays a role in social bonding and attachment [9].

All in all, the delicate balance of neurotransmitters such as dopamine, serotonin, and glutamate, and hormones such as those mentioned above, is essential for maintaining normal behavioral functions. Disruptions in this balance can thus result in altered behavior, as seen in various psychiatric disorders [10].

1.1.2 **A brief history of psychiatry**

The study of these altered behaviors, encompassed by the field of psychiatry, has a rich and storied history, marked by evolving theories and approaches to understanding and treating altered behavior in the context of mental health. The beginnings of psychiatry can be traced back to ancient civilizations, where mental disorders were often attributed to supernatural forces or divine intervention [11]. However, the modern understanding of psychiatry truly emerged during the Age of Enlightenment in the 18th century, when the focus shifted towards a more scientific and humane approach to mental health [12].

One of the pioneers of this era was Philippe Pinel, a French physician who advocated for a compassionate approach to treating individuals with mental disorders. He emphasized the importance of understanding the root causes of altered behavior, paving the way for the development of modern psychiatric theories and therapies [13]. As psychiatry evolved throughout more recent periods of history, the field expanded its knowledge of the underlying biological processes influencing behavior, drawing upon the aforementioned discoveries in genetics, neurobiology, and endocrinology.

The 20th century marked significant advances in psychiatric research and treatment, driven by the emergence of psychoanalysis, behaviorism, and psychopharmacology. Sigmund Freud's psychoanalytic theory, which focused on unconscious processes and internal conflicts, had a profound impact on the understanding of human behavior [14]. Simultaneously, behaviorism, led by figures such as John Watson and B. F. Skinner, emphasized the role of observable behaviors and environmental influences in shaping human behavior [15]. Psychopharmacology, the study of how drugs affect the mind and behavior, opened new avenues for treating psychiatric disorders by targeting the imbalances in neurotransmitters and hormones associated with them [16].

Despite the progress made in understanding and treating psychiatric disorders throughout history, however, challenges remain in fully elucidating the complex biological processes underlying altered behavior [17]. To address these challenges, researchers have increasingly turned to animal models as invaluable tools for studying the genetic, neurobiological, and physiological aspects of behavior [18, 19, 20, 21, 22]. These models provide controlled environments in which researchers can manipulate specific factors, such as genetic mutations or environmental stressors, and measure their impact, typically in specific relevant variables [23].

Animal research has thus yielded essential insights into the neurobiology of psychiatric disorders, such as the role of neurotransmitter systems, neural circuitry, and genetic factors in the manifestation of altered behavior. For instance, rodent models have been crucial in understanding the role of dopamine in reward-related behaviors and addiction [24], as well as the involvement of serotonin in mood regulation and the pathophysiology of depression [25]. Additionally, animal models of stress have helped to elucidate the biological underpinnings of stress-related psychiatric disorders, such as anxiety and post-traumatic stress disorder (PTSD) [21, 22]. These findings highlight the importance of accurate behavioral quantification in understanding the etiology and progression of psychiatric disorders, as well as in the development of novel therapeutic strategies.

1.1.3 Bridging the translational gap: from animal models back to humans

While animal models have been instrumental in advancing the understanding of the neurobiology of psychiatric disorders, a translation gap persists when applying these findings to improve the lives of human patients. This gap arises from various factors, including differences in species, the complexity of human behavior, and the limitations of animal models in capturing the full spectrum of psychiatric symptoms [23, 26]. In addition, the often fuzzy symptom-based definition of psychiatric disorders in humans makes it hard to disentangle unique biological mechanisms underlying disorders that fall into the same classification [27]. Along these lines, initiatives such as the Research Domain Criteria (RDoC) have opened the field for discussion about more comprehensive, multimodal definitions of psychiatric disorders, which could have a positive impact in the future [28].

Thus, one of the primary challenges in bridging the aforementioned translation gap relates to the inherent differences between species. Although rodent models share some genetic, neurobiological, and physiological similarities with humans, there are significant differences in brain structure, function, and complexity. Consequently, the behavioral responses and underlying neurobiological mechanisms observed in animals may not fully mimic those in humans [29].

Moreover, human behavior is shaped by a multitude of factors, including culture, personal experiences, and social interactions [30]. Animal models, although useful for studying basic biological processes, may not adequately capture the intricacies of human behavior and the unique environmental contexts that influence it. For instance, animal models of depression may rely on stress-induced behaviors, but these may not encompass the full range of cognitive and emotional symptoms experienced by humans with depression [31].

Additionally, the validity of animal models in psychiatry depends on their ability to accurately mimic the clinical features of psychiatric disorders. While some animal models have been successful in recapitulating certain aspects of human disorders, they often do not cover the entire spectrum of symptoms or the heterogeneity observed in clinical populations. This limitation can hinder the development of effective treatments that address the diverse presentations of psychiatric conditions [23, 30].

To minimize this translation gap, researchers are continuously refining animal models and developing new experimental paradigms that better reflect the complexity of human behavior and psychiatric symptoms. Along these lines, recent models, such as depression-like syndrome (DLS) in mice (Figure 1.1), leverage new technologies on comprehensive measuring to mix clinical criteria and RDoC to provide bio-behavioral reference syndromes for preclinical rodent models [31].

In parallel, the evolution of behavioral quantification has played a pivotal role in advancing psychiatric research using both animal models and patients. From classical ethology to virtual reality and multi-modal tracking, the next sections explore the science behind translating such a complex phenomenon as behavior into meaningful quantitative variables.

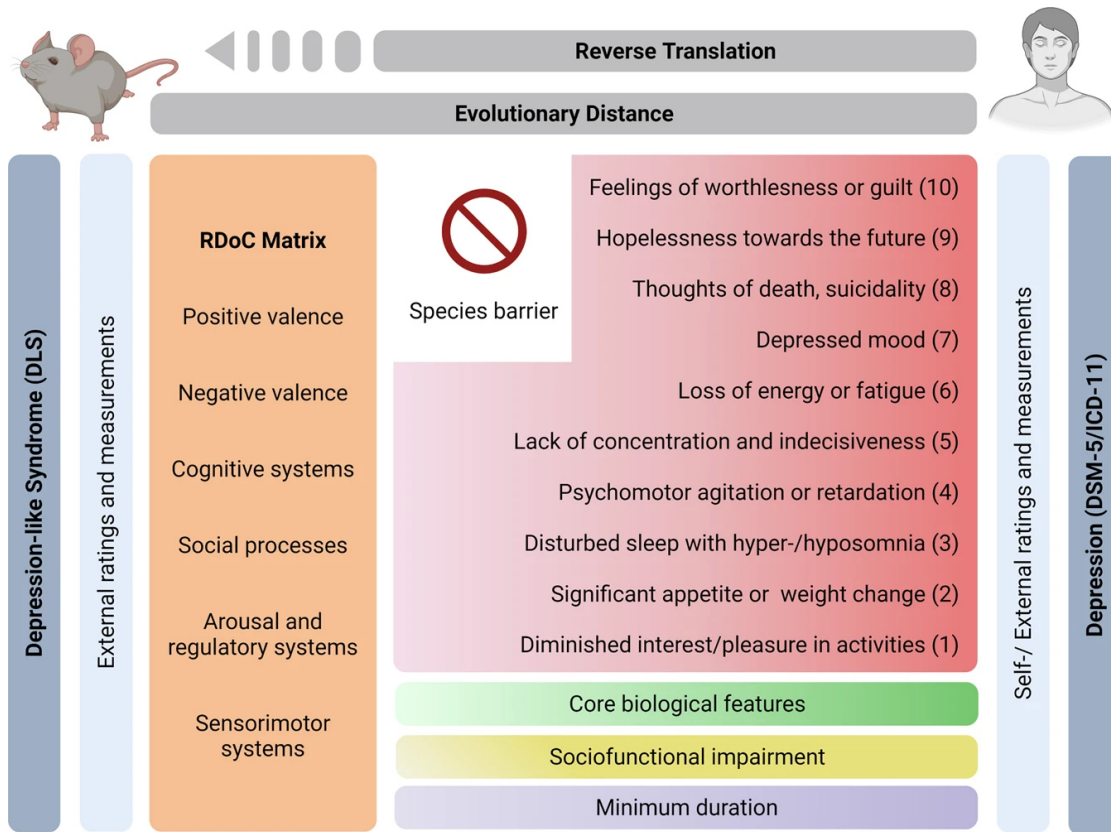


Figure 1.1: Depression-like syndrome in mice: Presented by von Mücke-Heim et al. in 2022, this model aims to bridge the gap between murine and human depression models. Beginning with the DSM/ICD definition of depression, the provided matrix illustrates the process of reverse translation, which takes criteria from human clinical settings and seeks to apply them to mice. There are, of course, certain symptoms of depression, such as feelings of worthlessness, that cannot be translated due to the evolutionary distance and species-specific barriers. However, many other symptoms, as well as key biological markers, socio-functional issues, and a certain minimum duration, can be effectively measured in mice. Examples of these include a decrease in appetite or significant weight loss. Both human and mouse measurements can then be sorted into RDoC domains. (Created with BioRender.com, adapted from [31]).

1.2 Quantifying behavior

The field of animal behavior quantification has evolved considerably since its early days in the discipline of ethology. Pioneers such as Konrad Lorenz and Niko Tinbergen laid the foundation for systematic observations of animals in their natural environments, helping to establish fundamental principles and patterns of behavior [32]. As the field progressed, researchers began to conduct controlled experiments in laboratory settings, allowing for more precise measurements and manipulations of experimental variables [33]. These

1 Introduction

advancements in methodology provided valuable insights into the underlying mechanisms of animal behavior, which have in turn informed the development of novel technologies and computational techniques for quantifying and analyzing complex behavioral data [34].

1.2.1 Endless forms most beautiful: from ethology to controlled behavioral experiments

When the brain encounters external stimuli, it triggers specific patterns of cellular responses that ultimately shape behavioral outcomes. In the past, the foundational principles of behavior were typically derived from observational studies, where animals were left undisturbed in their natural environments [32]. Early influential work in this area was carried out by Charles Darwin in the 19th century. In his seminal work, “On the Origin of Species” (1859) [35], he introduced the concept of natural selection, which provided an explanation for the diverse array of shapes and behaviors observed in the animal kingdom. Moreover, his book “The Expression of the Emotions in Man and Animals” (1872) [36] further delved into the subject, investigating the evolution and adaptive value of emotional expressions in both humans and animals. Thus, Darwin’s work established a foundation for subsequent ethologists, arguably shaping the discipline’s core principles.

Thus and so, ethology was set to primarily work through observational methods, a core assumption of the discipline being that the most comprehensive understanding of behavior can be achieved by observing animals in natural or semi-natural environments. This approach allows researchers to study behavior descriptively, generating hypotheses and uncovering new behavioral concepts [37]. A notable example of exceptional ethological achievement is the work led by Konrad Lorenz (1903–1989), an Austrian zoologist who is best known for his research on imprinting, a rapid learning process that occurs early in an animal’s life, during which it forms strong attachments to certain stimuli. Through his work with greylag geese, Lorenz discovered that newly hatched goslings would imprint onto him, treating him as their parent. This observation revealed the innate nature of certain behavioral patterns, and emphasized the role of critical periods in the development of species-typical behaviors [38]. Along these lines, Niko Tinbergen (1907–1988), a Dutch biologist, set the grounds for a more comprehensive understanding of animal behavior with his “four questions” framework, which is used to analyze animal behavior from four different perspectives: mechanism (or proximate causation), function (or ultimate causation), ontogeny, and phylogeny [39]. Ultimately, Tinbergen and Lorenz were awarded the Nobel Prize in Physiology or Medicine in 1973 (along with Karl von Frisch) for their discoveries concerning animal behavioral patterns.

While ethology and observational research have produced remarkable findings, however, there are significant limitations to these study designs. For instance, observational studies depend on the researcher’s ability to accurately assess behavior, which can result in misinterpretation and differing interpretations among researchers. Additionally, the absence of control over environmental variables can lead to poorly reproducible results due, for example, to high variability between experimental conditions.

To address these limitations, the field of comparative psychology emerged, where researchers sought to uncover general principles of learning, cognition, and behavior within tightly controlled environments [33]. The emphasis then shifted towards deconstructing the overall behavior into distinct, measurable elements, thus reducing its complexity through controlled laboratory settings, and enabling the isolation of the effects of specific factors on behavior (Figure 1.2). These lab tasks are characterized by their high degree of environmental control and standardized behavioral readouts, making causal inference and hypothesis-driven research questions possible [40]. Pioneering researchers like Edward Thorndike, Ivan Pavlov, Burrhus Frederic Skinner, and others demonstrated the potential and power of this field. Along these lines, Thorndike illustrated the concept of trial-and-error learning by showing that animals became more efficient at escaping a device and obtaining rewards with an increasing number of trials [41]. Skinner, subsequently, developed one of the earliest and most popular laboratory behavioral tasks, the operant conditioning chamber (or “Skinner box”), which can be used for both negative and positive reinforcement learning and still remains widely used in research [42]. This development initiated a trend to standardize and simplify various behavioral disciplines using laboratory tasks, a trend that continues to this day (Figure 1.3). Laboratory tasks are of undeniable value for investigating the effects of external stimuli (e. g., the stress response system) and interventions (e. g., pharmacological) on behavior. Furthermore, the unparalleled possibilities of using various genetic mouse models have facilitated the study of specific target genes on behavior [23].

However, no behavioral tasks are flawless, and they carry assumptions and particular concerns that must be addressed. First and foremost, the general laboratory setups involve intensive interaction between the researcher and the test animals, raising concerns that inter-individual differences among researchers, such as sex, could impact the animals’ behavioral performance [43, 44, 45]. Moreover, the current laboratory housing settings are highly unnatural, preventing rodents from engaging in species-typical behaviors and causing problematic behavior such as extreme aggression in group-housed animals [46]. Lastly, the use of inbred mice presents challenges when investigating naturalistic behaviors. Although genetic models have provided valuable insights into the genome, they have also resulted in animal models that behave quite differently from their wild counterparts, calling into question the validity of these models and, unfortunately, reducing the reproducibility of behavioral research [30].

All in all, taking a reductionist approach in laboratory tasks can be advantageous for many behavioral disciplines; however, it may also be detrimental for behavioral constructs that rely on multiple outputs and more naturalistic environments, making them more complex to evaluate. Recently, new technologies have enabled researchers to track animals in semi-naturalistic open-field settings in decreasingly invasive ways. By extracting reliable information from less restricted environments, experimenters can increase efficiency (by yielding multiple automatic read-outs per experiment) and explore more natural settings while retaining control over experimental variables (such as genetics or drug administration). Along these lines, the next section will delve into the computer science and machine learning (ML) advances that enabled this trend, and how they came to be.

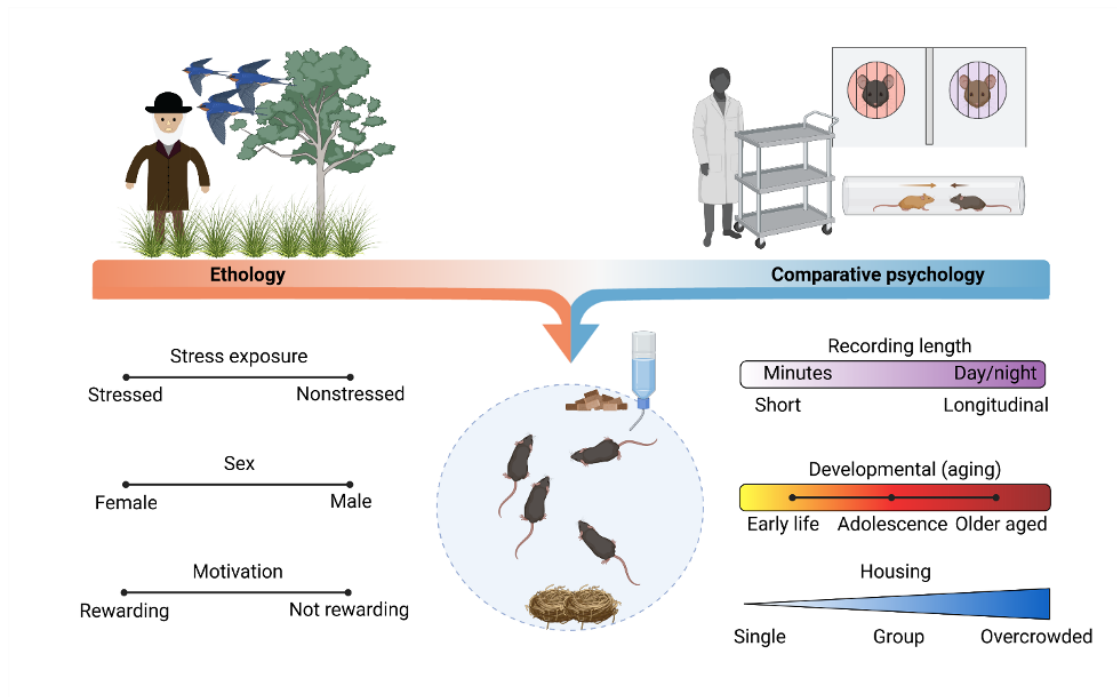


Figure 1.2: From ethology to modern behavioral quantification: The upper section of the figure depicts key methodologies used in behavioral neuroscience research, showing ethology on the left and comparative psychology on the right. The next generation of social behavioral tests, signified by the central arrow, utilize a semi-naturalistic environment. This combines the strengths of both ethology, providing a naturalistic setup free of experimenter interference, and comparative psychology, maintaining some level of environmental control by restricting space and external influences. Various elements, including exposure to stress, gender, motivation, recording duration, age-related changes, and living conditions, will have an impact on the results of the social behavioral evaluation. (Adapted from [47]).

1.2.2 Deep learning and the advent of markerless pose estimation

The idea of tracking animals in more naturalistic experimental settings is not new. For example, in 2013 Shemesh et al. created an automatic phenotyping system based on video color recognition called the “Social Box” [48]. The authors described how social behavior in mice develops in a semi-natural environment, using techniques that quantify behavioral traits automatically, thus liberating researchers from laborious manual quantification. The authors automatically tracked several groups of mice in their home environment and investigated how individual behavior is strongly interdependent in their groups. In a follow-up study in 2019, Forkosh and colleagues developed a model, using the same system, that captures and outlines stable personality traits in mice [49]. While insightful, this work and subsequent studies were limited to tracking each animal’s central position. Moreover, in these and other contemporary approaches, animal identification relied on dedicated (and often expensive or invasive) physical markers, such

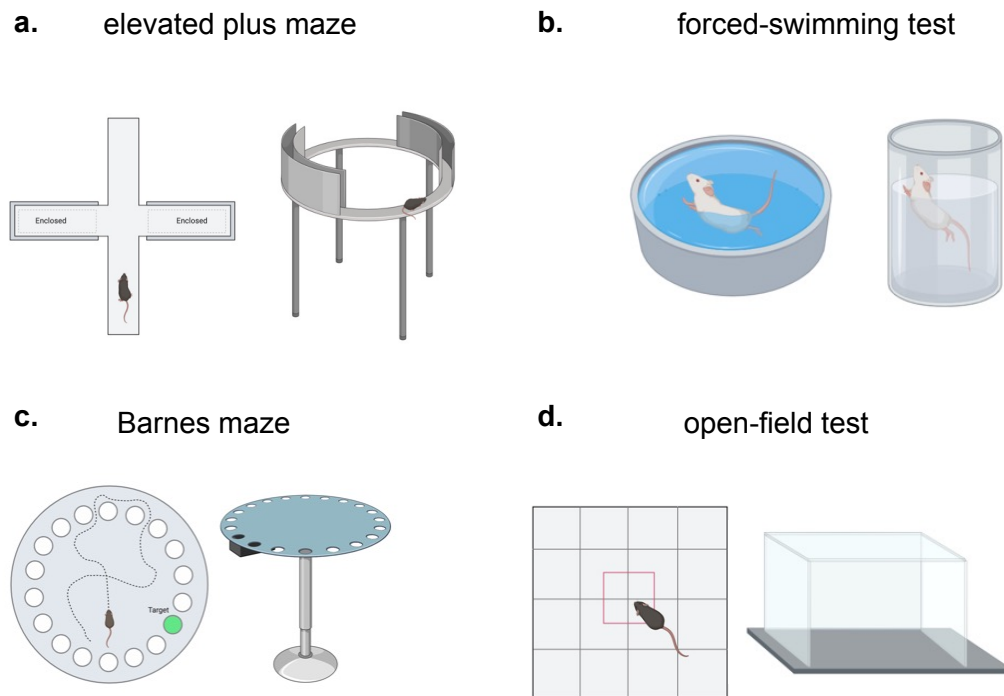


Figure 1.3: Common univariate laboratory behavioral tasks: **a.** Elevated plus mazes are commonly used setups to test for anxiety-like behavior. As anxious animals will tend to spend less time in the open arms, the ratio between time spent in open versus closed arms is typically reported. **b.** Forced-swimming tests are often used to measure anhedonic-like behavior, which is reported through helplessness time (the time the animals spend without trying to actively leave the pool where they are submerged). **c.** Barnes' mazes are common tools to measure memory in rodents, where their ability to remember the location of specific target zones is tested. **d.** Open-field tests, typically used to assess locomotion, are becoming increasingly popular for automated feature extraction following pose estimation. (Created with BioRender.com).

as radio frequency identifiers (RFID) or color hair dyes. Over the last decade, however, advancements in computer science, especially machine learning and deep learning-based computer vision, have sparked a revolution in animal motion tracking, enabling non-invasive markerless pose estimation of multiple body parts, which opened the floor for innumerable and creative ways of extracting behavioral information from more naturalistic (albeit controlled) environments.

To understand how this came to be, let us start from the beginning. Machine learning, a subfield within artificial intelligence (AI) which deals with methods that leverage data to improve computer performance on some set of tasks [50], arguably has its origins in the mid-20th century. One of the earliest proposed algorithms was the perceptron, introduced by Frank Rosenblatt in 1957 [51], which could be seen as a simple linear classifier

1 Introduction

that could learn to recognize patterns in data. Although limited in its capabilities, it laid the foundation for more advanced techniques, such as support vector machines and more complex neural networks. The 1980s subsequently marked the beginning of the modern era of machine learning with the emergence of decision trees, k-nearest neighbors (KNNs), and other algorithms that enabled computers to learn from data more effectively [52]. Furthermore, the development of the backpropagation algorithm by Geoffrey Hinton and his colleagues in 1986 allowed for more efficient training of neural networks, setting the stage for the rise of deep learning [53].

Deep learning is born then as a subfield of machine learning, which involves the use of artificial neural networks (ANNs) with multiple hidden layers to learn complex, hierarchical representations of data [50]. The first breakthrough of deep learning arguably came in 2012 when Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton developed AlexNet, a deep convolutional neural network (CNN) that significantly outperformed other methods in the ImageNet Large Scale Visual Recognition Challenge [54]. This success marked the beginning of the “deep learning revolution” and led to rapid advancements in various AI applications, including computer vision, natural language processing, and speech recognition. In computer vision in particular, deep learning enabled the departure from hand-crafted feature extraction methods popular in since the 1960s, which proved to be insufficient for complex, real-world tasks [55]. Since then, there have been numerous breakthroughs, including Faster R-CNN for object detection [56], Generative Adversarial Networks (GANs) and diffusion models for image synthesis [57, 58], and the introduction of the Transformer architecture, which has further enhanced the capabilities of natural language processing and computer vision systems [59], among others. Moreover, the development of increasingly performant models for multi-class image classification led to the rise of effective transfer learning, where models that had been pre-trained in large datasets, such as ImageNet, can be used for feature extraction, and fine-tuned or repurposed altogether for a different downstream task, without the need for full retraining [60].

It is in this context that tools like DeepLabCut [61], SLEAP [62], and SIPEC [63], were developed in the last few years. By leveraging deep neural networks (DNNs) pre-trained in ImageNet, base versions of these models are capable of detecting the position of a set of user-defined body parts in each frame of a given video dataset, upon fine-tuning with very little human labelling. While several architectures have been developed to date, the basic idea consists of replacing the classification layers of a chosen pre-trained model (typically ResNet50 [64]) with deconvolutional layers that output a probability mass map for each body part, and each pixel on the original image (Figure 1.4). Furthermore, the *argmax* of the output for each body part can then be interpreted as a confidence value, enabling further downstream filtering or processing of defective tracks. The incorporation of deep learning techniques into animal motion tracking has not only simplified the data collection process, but also improved the quality and granularity of the information gathered, providing researchers with unparalleled opportunities to investigate intricate behavioral patterns and their underlying neural mechanisms, both while retaining control on experimental variables and in the wild [65].

In this fashion, these approaches have made it possible to gather vast amounts of time series data on multiple body parts with human-level accuracy [66]. Additionally, some models can now retain individual identification in social settings without dedicated hardware, making it possible to track multiple animals at once, which paves the way for social behavioral analysis [62, 67].

Moreover, and in contrast to instances where experimental breakthroughs have triggered an increase in data volume (thereby sparking the need for new computational approaches) the case of behavioral analysis has followed the opposite trend [68]. Here, the deliberate application of recent computational techniques has led to a rapid increase in data collection, enabling even more technical breakthroughs (such as foundation models for particular species [69]), and ultimately changing how research is conducted and the types of questions people can ask [34]. In the next section, we discuss how precision tracking data can be analyzed, to gain new insights into animal behavior and answer scientific questions that were much harder to address before this field came to be.

1.3 Automated annotation of motion tracking data

As previously discussed, deep learning based pose estimation made a strong impact in the amount of information that can be extracted from raw animal experiment video. In this section, we will explore the plethora of methods that became available to annotate, analyze, and ultimately extract meaning from this novel and rich paradigm. These methods will be mostly described as falling into one of two big families, namely supervised classification (aiming to extract pre-determined and characterized traits) and unsupervised embedding and clustering (seeking to explore data and extract patterns without explicit external input). Combinations between the two, such as clustering-powered active learning platforms, will also be touched upon.

1.3.1 Supervised annotation

Either by observational research or as the result of controlled experiments, behavioral neuroscience has had time to develop and describe detailed ethograms for specific animal models [70]. That is, descriptions of typical actions these experimental animals conduct in the environments in which they are typically observed. The supervised annotation of motion tracking time series, then, aims to use classification models to detect, upon being fed with labelled examples, the moments in time where animals perform certain actions, that may or may not be related to a specific experimental condition.

Along these lines, a current common-use package that requires minimal coding to create supervised behaviors from DLC and other pose-estimation packages is SimBA (Simple Behavioral Analysis [71]). SimBA enables users to label behaviors of interest in a graphical user interface, training machine learning models that can learn the rules underlying certain behaviors directly from data, thus automating the quantification of arbitrarily complex traits. Moreover, the provided models (typically ensemble classifiers such as Gradient Boosting Machines—GBMs) are trained on extracted static and dynamic features describing animal motion, rather than the sequences themselves [72].

1 Introduction

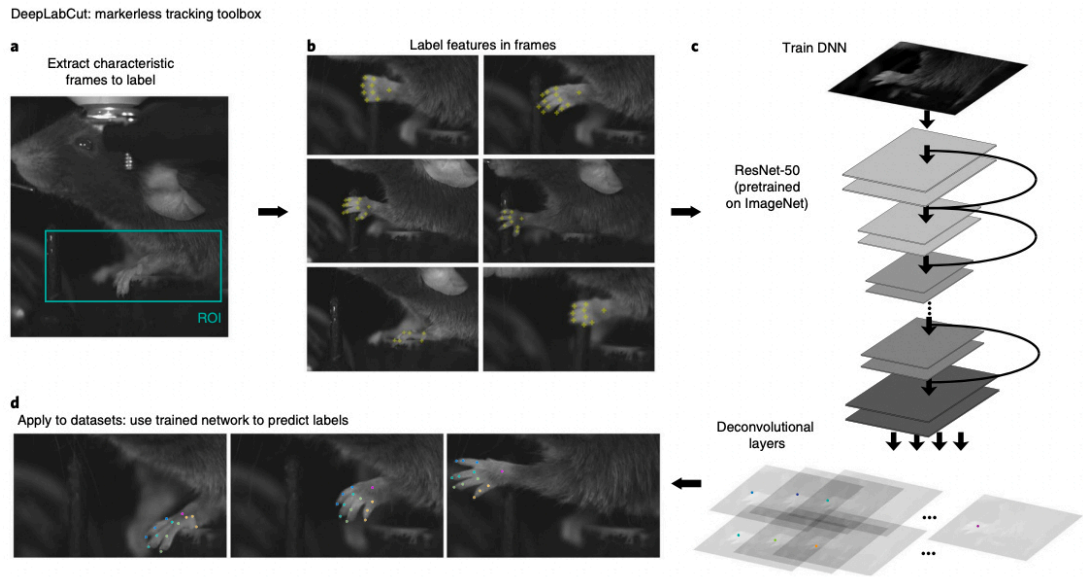


Figure 1.4: DeepLabCut overview: **a.** The process begins with the extraction of images displaying distinct postures that are representative of the specific animal behavior. To enhance computational efficiency, the region of interest (ROI) should be minimized, while still encompassing the behavior under study, which in this case is reaching. **b.** Next, the user is required to manually identify and label various body parts. In this context, different joints of the digits and the wrist were pinpointed as features of interest. **c.** A deep neural network (DNN) architecture is then trained to predict the locations of the labeled body parts based on the associated image. A unique readout layer is produced for each body part, designed to predict the likelihood of a body part appearing in a specific pixel. Training adjusts the readout and DNN weights, which are stored post-training. **d.** The trained network can then be utilized to determine the positions of the body parts from video footage. The images depict the most probable locations for the 13 labeled body parts on a mouse’s hand. (Adapted from [61]).

This approach makes the models lighter to train and doesn’t require a dedicated graphics processing unit (GPU). Additionally, certain packages, such as the previously mentioned SIPEC [63] or MARS [73], integrate a custom tracking system with a supervised behavioral annotation pipeline, offering benefits over combining different packages for the same purpose since users do not need to worry about software compatibility within different frameworks.

All these approaches work very well at detecting previously defined patterns. However, their generalizability across behavioral setups is rather limited for complex behaviors, forcing users to relabel data and retrain models almost every time a new dataset comes along. Nonetheless, some behaviors can be accurately reduced to a set of hard-coded rules, without the need for model training [74]. These include, but are not limited to, time-in-zone quantification, specific individual interactions (e. g., nose to nose and nose

to tail contacts), or interactions with objects. Simple approaches in this case can help reduce overfitting in some very relevant scenarios [75]. Moreover and leaving generalizability issues aside, while supervised classification models can drastically increase annotation throughput when compared to classical univariate behavioral tasks, they are still restricted to a set of pre-defined and labelled patterns. This can result in leaving out meaningful yet undescribed animal behaviors, which could be better captured with a different set of methods.

1.3.2 Unsupervised annotation and behavioral embeddings

An alternative approach to annotating behavior involves analyzing pose-estimation data without pre-established ethograms. This can be achieved through unsupervised learning techniques, a branch of machine learning that seeks to extract insights from data without prior information about the outcomes of interest [52].

By retrieving behavioral clusters or syllables expressed at particular points in time, unsupervised algorithms in this context have the potential to help create automatic comprehensive ethograms, bypassing the limitations of the previously described classification methods, and hinting by themselves at new knowledge [76].

Additionally, these approaches facilitate hypothesis generation, as they can identify behavioral patterns indicative of previously unknown behaviors within a specific context. Since unsupervised methods allow for exploration of the behavioral space without the need for labeling, they can serve as an initial screening for behaviors of interest. For instance, clustering can be employed to pinpoint at particular behavioral patterns displaying the most significant differences between predefined experimental conditions [75]. Subsequently, researchers can train supervised classifiers to more directly and accurately measure the behaviors of interest. In this regard, new approaches have been published recently in which unsupervised clusters are used to initialize supervised classifiers, which are in turn fine-tuned using an active learning framework [77].

Along these lines, several packages and pipelines have been developed that automatically segment behavior using unsupervised methods. For example, the software system B-SOiD [78] annotates motion data with feature sets that help describe behavior over time without directly using sequential data. Another software system, MoSeq [76, 79], leverages the time component of motion with autoregressive hidden Markov models, which can directly capture probabilistic relationships between input variables. Other packages, like VAME [80], employ neural networks that process motion sequences directly, and apply post-hoc clustering techniques on the resulting latent space, yielding more meaningful clusters of specific behaviors with less noise from other patterns [81]. Thus and so, an added benefit of neural network models is their ability to embed motion trajectories into interpretable latent spaces that can be, for example, analyzed differentially across experimental conditions [75]. Moreover, understanding the retrieved patterns is crucial for comprehending the underlying behaviors. The ability to interpret the models' outputs can not only enhance transferability across datasets in the case of supervised learning, but also aid the meaningful interpretation of unsupervised clusters. In this context, both visual exploration (taking advantage of video data and mapping

1 Introduction

back to snippets assigned to specific patterns) and machine learning explainability tools, such as Shapley Additive Explanations (SHAP) [82], are suitable approaches.

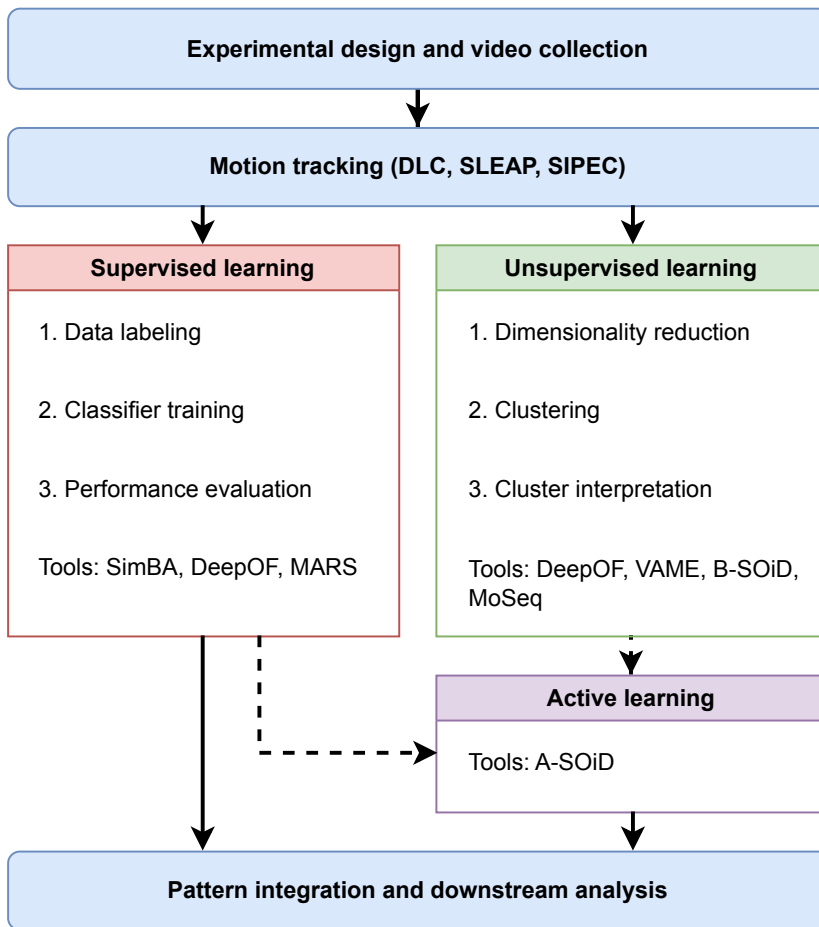


Figure 1.5: Automatic behavioral annotation via motion tracking: Once the experimental design is established and videos are gathered, keypoints from one or multiple animals are extracted over time as time series, using tools like DeepLabCut, SLEAP, or SIPEC. Predetermined behaviors can be identified with supervised learning tools such as SimBA, or MARS, which typically require data labeling, classifier training, and an evaluation of the performance of the extracted behaviors. However, tools like DeepOF offer pre-trained models, eliminating these steps. An alternative method for obtaining a wider scope of information is through unsupervised learning, which doesn't require labeling, and aims to derive behavioral syllables or clusters following a dimensionality reduction step. The interpretation of these clusters is also a crucial step, which can be done via visual video inspection or using model explainability techniques like SHAP. This category includes tools such as DeepOF, VAME, B-SOiD, and MoSeq, to name a few. In addition, the results from unsupervised learning can serve as a starting point for supervised models with active human feedback, as demonstrated in the A-SOiD framework. Lastly, the expression and dynamics of all acquired patterns can be compared across different experimental conditions to gain insights into behavioral changes. (Adapted from [47]).

1 Introduction

In summary, the use of supervised and unsupervised methods in automated behavioral analysis is key to leverage new tracking technologies, and holds the potential to significantly contribute to our understanding of animal behavior in the near future, revealing new patterns and helping to streamline research efforts. New and innovative methods that can help analyze these data in meaningful ways can thus positively impact the current state of the field. An overview of this pipeline can be found in figure 1.5.

The next section of this introduction will delve into the fundamental biology and behavioral implications of chronic stress, which serves as the case study for applying the algorithms and analyses developed in this thesis.

1.4 Chronic stress as a case study

Chronic stress serves as an ideal illustration of the complex connections between animal behavior and psychiatric disorders. It has been associated with the development or worsening of numerous psychiatric conditions, including major depressive disorder (MDD), post-traumatic stress disorder (PTSD), anxiety disorders, and even neurodegenerative diseases like Alzheimer’s and Parkinson’s [83]. In animal models, chronic stress exposure can lead to behavioral changes similar to those observed in human patients, such as heightened anxiety-like behavior, cognitive impairments, and disruptions in social interactions [84, 85, 86]. Examining the effects of stress on animal behavior has not only deepened our understanding of the intricate relationship between stress and psychiatric disorders but has also provided insights into underlying neurobiological mechanisms and potential therapeutic interventions. In this section, we explore the basic biology behind stress, and the Chronic Social Defeat Stress (CSDS) model in mice, which will serve as a case study for the models presented in this thesis.

1.4.1 A primer on stress biology

Stress is an integral part of our daily lives, affecting our mood and motivation. Biologically speaking, it refers to the body’s response to external or internal stimuli that challenge its equilibrium, known as stressors. This response involves the complex interplay of neural and endocrine structures to help the organism adapt to particularly demanding situations. Along these lines, central to the stress response is the activation of the hypothalamic-pituitary-adrenal (HPA) axis, which leads to the release of cortisol, a hormone central to energy, blood pressure, and mood regulation. As a consequence, the sympathetic nervous system (SNS) is activated, resulting in the secretion of catecholamines, such as adrenaline and noradrenaline, from the adrenal medulla [87]. These neuroendocrine changes prepare the body for the “fight or flight” response, a concept introduced by Walter Cannon to describe the physiological adaptations that enable an individual to confront or evade a perceived threat [88]. This response includes increased heart rate, blood pressure, and respiration, as well as heightened alertness and the mobilization of energy resources. While short-term activation of the stress response can be beneficial for survival, chronic stress can lead to detrimental effects on physical and

mental health, highlighting the significance of understanding the biological mechanisms underlying its regulation [85].

Notably, chronic stress has increasingly become a societal burden, with the incidence of stress-related disorders steadily growing over the past decades [89]. Furthermore, our understanding of the behavioral and neurobiological mechanisms related to these disorders is limited, which contributes to the moderate success of current drug treatments [90]. Furthermore, the adopted symptom-based classification of these disorders, and the resulting heterogeneity, makes it often difficult to uncover their potential common biological causes [91, 27].

Along these lines, deciphering the complexity of neurobiological circuits and molecular pathways underlying healthy or abnormal stress responses requires the integration of cellular, molecular, and behavioral data [68] (Figure 1.6). While traditional approaches may lack the necessary spatial and temporal resolution, recent technological advancements have considerably improved these aspects. For example, single-cell transcriptomics enabled the investigation of thousands of genes simultaneously and the dissection of the contributions of different cell types involved in the stress response [92]. Similarly, the use of activity-dependent labeling methods combined with brain-clearing techniques allows for the identification of activated cells following specific stressors and the reconstruction of the involved brain circuits in a particular stress response [93]. As with most high-throughput techniques, these strategies generate vast amounts of data, necessitating the use of appropriate computational and statistical tools. Consequently, advancements in molecular and cellular neuroscience techniques have spurred growth in computational science and the development of suitable data analysis software. This way, the previously discussed novelties in behavioral phenotyping have also enabled researchers to efficiently assess the specific effects of different types of stressors, stress paradigms, developmental ages, and sex on behavior, while significantly reducing manual scoring-related bias [90]. Moreover, advances in virtual reality (VR) have also allowed researchers to test therapies and track behavioral responses in human patients [94, 95].

Furthermore, and as previously discussed, to understand the cellular and molecular mechanisms responsible for the pathophysiology of psychiatric disorders, it is crucial to develop and implement preclinical animal models. A common model used in current neurobiological research, namely Chronic Social Defeat Stress (CSDS) is presented in the next section.

1.4.2 Chronic Social Defeat Stress (CSDS)

In this context, the CSDS paradigm is a widely used animal model, predominantly in rodents, for studying the effects of chronic stress on behavior and its potential contribution to the development of stress-related psychiatric disorders, such as MDD, anxiety disorder, and PTSD [96]. The model aims to simulate territorial dominance, and involves repeated exposure to social stress through daily confrontations between a test subject, typically a mouse, and a more aggressive and dominant conspecific [97]. These encounters generally include physical aggression, which leads to subordination and submission in the test animal. Thus, and while far from perfect, the CSDS paradigm is designed

1 Introduction

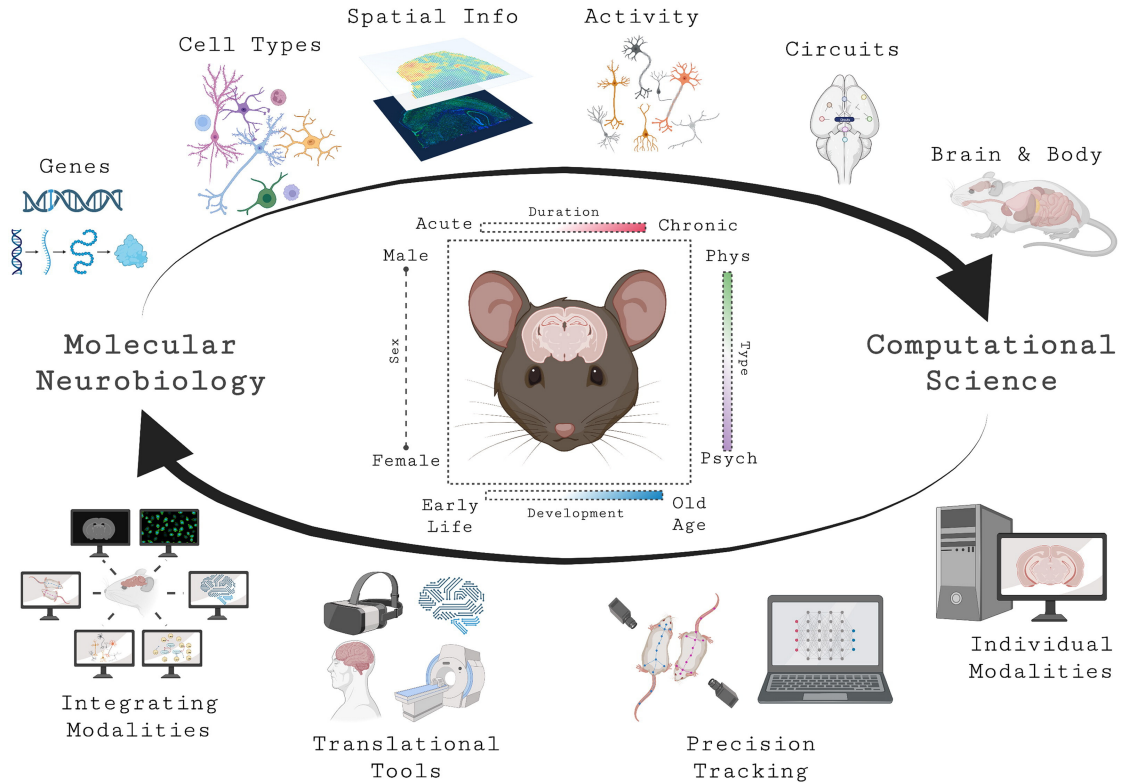


Figure 1.6: From macro to micro: increasing resolution in stress neurobiology. The field of stress neurobiology is benefiting from the increasing resolution offered by cutting-edge molecular and computational advancements. This focus on enhanced spatiotemporal resolution is affecting cell biology, behavioral science, and their interaction. The stress response can be studied from various perspectives, such as the nature of the stressor (physical versus psychological), the duration of the stressor (acute or chronic), the stage of development (from early life through adolescence, adulthood, and old age), or even the sex of the subject (male or female). (Created with BioRender.com, adapted from [68]).

to mimic aspects of chronic stress experienced by humans and induce behavioral, neurobiological, and physiological changes that resemble those observed in stress-related psychiatric conditions.

In most commonly used CSDS protocols, test animals undergo a series of daily stress exposures, usually lasting between 10 and 21 days [97]. After each confrontation with the conspecific, the test subject and the aggressor are housed in the same cage but separated by a mesh-like barrier, allowing continuous sensory contact while preventing further physical aggression. This continuous exposure to the stressor promotes the development of stress-related behaviors in the test animal, such as anhedonia, anxiety-like behavior, reduced motivation, and social avoidance [98, 99, 100]. Following the chronic stress exposure, researchers typically assess these behaviors using various tests, aiming to quantify differences in social interaction and avoidance, sucrose preference, or locomotion

(among others), to evaluate the effects of CSDS on the animal. Additionally, and in line with what was previously discussed about animal models, the CSDS paradigm enables the investigation of the underlying neurobiological and molecular mechanisms involved in stress-induced behavioral changes, as well as the evaluation of potential therapeutic interventions for stress-related disorders [101, 102].

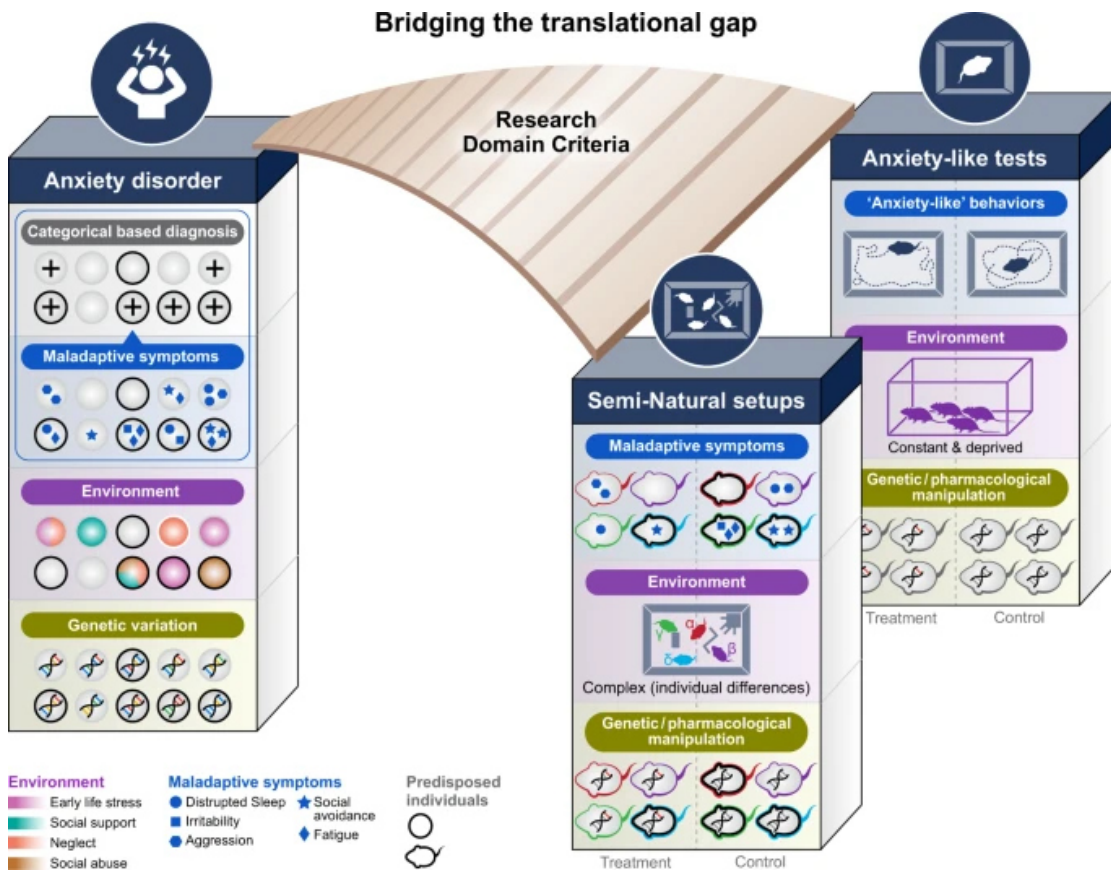


Figure 1.7: A paradigm shift in translational psychiatry through rodent neuroethology: So far, the use of animal models in the translational research of mental disorders hasn't managed to live up to the hopes of discovering new treatments. Human mental illnesses like anxiety disorders are complex, shaped by genetic factors, environmental influences past and present, and the interplay between these elements. In this context, disease diagnosis relies on categorized behavioral criteria (indicated in the figure by '+' symbols). In contrast, constructs like anxiety in mice are typically modeled using brief tests that identify 'disease-like' phenotypes. Despite the high level of control, this behavioristic approach falls short in capturing the complexity of human mental illnesses. Automated tracking of complex behavior in mouse groups (in a semi-natural setup) can disclose individual coping mechanisms, so that the integration of univariate tests and semi-natural setups in the study of endophenotypes shared across mammalian species holds promise for narrowing this existing gap. (Adapted from [90]).

1 Introduction

While a widely adopted framework, however, the reduction of the overall shifts in behavior induced by this protocol can lead to an oversimplification of the associated behavioral repertoire, as well as to increasing the risk for cross-over effects with other types of behavior, such as anxiety. Moreover, due to technological limitations, the analysis of the interaction between multiple freely moving animals remained historically difficult, which further limited the complexity of the behavioral assessment [90]. Along these lines, social behavior is a complex entity that relies on many different types of behavioral interactions, which often are too complicated, time-intensive, and repetitive to assess manually [75]. Ultimately, this makes CSDS an optimal scenario for the development of new tools, since any newly developed methods should recapitulate the available knowledge (which acts as a positive control) and room for improvement both in terms of throughput and behavioral insight is clear. As previously discussed, and while it remains impossible to fully replicate human disorders in animal models, the systematic development and thorough characterization of existing and new models can not only benefit basic research, but also help to bridge the overarching translational gap that exists in psychiatry today [90] (Figure 1.7).

In this context, this thesis aims to develop and present novel methods to analyze and describe behavior in experimental mice, deploy them at scale, and apply them to increase resolution in current descriptions of CSDS. Before addressing these goals, chapter 2 will delve into the technical state of the art of the field.

2 State of the Art

2.1 Time series clustering

Aside from motion tracking, time series data is prevalent in numerous data mining applications, from weather forecasting to energy consumption predictions. In many scenarios, clustering techniques serve as a wonderful exploratory tool to group either similar time series altogether, or to obtain segments in single instances in which the data behaves in consistently similar ways [52].

In the general case, we consider a set of N objects represented as:

$$X = \{x_1, \dots, x_N\} \quad (2.1)$$

Taking a measure of dissimilarity between objects x_i and x_j that can be expressed as $d(x_i, x_j)$, the goal of clustering is to divide X into a partition $C = c_1, \dots, c_k$ consisting of K clusters, which maximizes both the similarity between objects within the same cluster, and the dissimilarity between objects in different clusters [103].

The overall objective of clustering in the context of this thesis, therefore, is to find a mapping function f_Θ that enables the obtaining of a relevant partition C for time series data. Unlike many other data types, however, time series have some peculiarities that make this problem exceptionally hard, and render traditional algorithms unsuitable without either pertinent modification or data preprocessing. To explore why this is the case, let us represent a time series of length T as:

$$x_i = [x_{i,1}, x_{i,2}, \dots, x_{i,T}] \quad (2.2)$$

Here, $x_i \in \mathbb{R}^{(d \times T)}$, where d is the number of features for each time step. Series with $d = 1$ are considered univariate, and series with $d > 1$ (as virtually all cases explored in this thesis) are deemed multivariate. That said, the unique nature of the time dimension presents challenges when employing traditional clustering methods. For starters, each time step cannot be regarded as an independent feature, with observations displaying varying degrees of autocorrelation [104]. Furthermore, two time series may represent similar objects, but their time signals could be delayed, stretched, or affected by noise (Figure 2.1). Consequently, these time series may exhibit significant differences in Euclidean space, despite representing similar signals [103]. As a result, researchers have proposed a plethora of time series-specific clustering methods in the literature.

Along these lines, more classical methods tend to rely on adapting time series to the Euclidean space through time-aware feature extraction [105], or to design alternative distance metrics that can take care of alignment issues [106]. While these methods are widely used for their simplicity and interpretability, they may struggle with high dimensional or noisy data (such as motion tracking), and can be computationally expensive for large datasets.

As an intermediate methods' family, probabilistic models, such as Hidden Markov Models (HMMs) retain some interpretability capabilities, while excelling at time segmentation even for time series with several dimensions. They typically exhibit difficulties handling long-range dependencies, however, due to the underlying Markovian assumption [104], and are in general less robust to noise than other alternatives, requiring specific tweaks tailored to each specific situation [76].

On the other hand, neural network-based methods have gained popularity for their ability to automatically learn complex features from the data [103]. These methods excel at handling high dimensional and noisy time series, and are capable of capturing long-range dependencies. However, they can be more challenging to interpret, require larger datasets for training, and may be prone to overfitting if not properly regularized.

In the following sections, we discuss in detail the primary time series clustering methods available in the literature, as well as their advantages and disadvantages in general and for motion tracking data in particular. A set of specific behavioral segmentation examples across many of these categories are provided, and the main ideas to explore in the algorithms presented in this thesis are introduced.

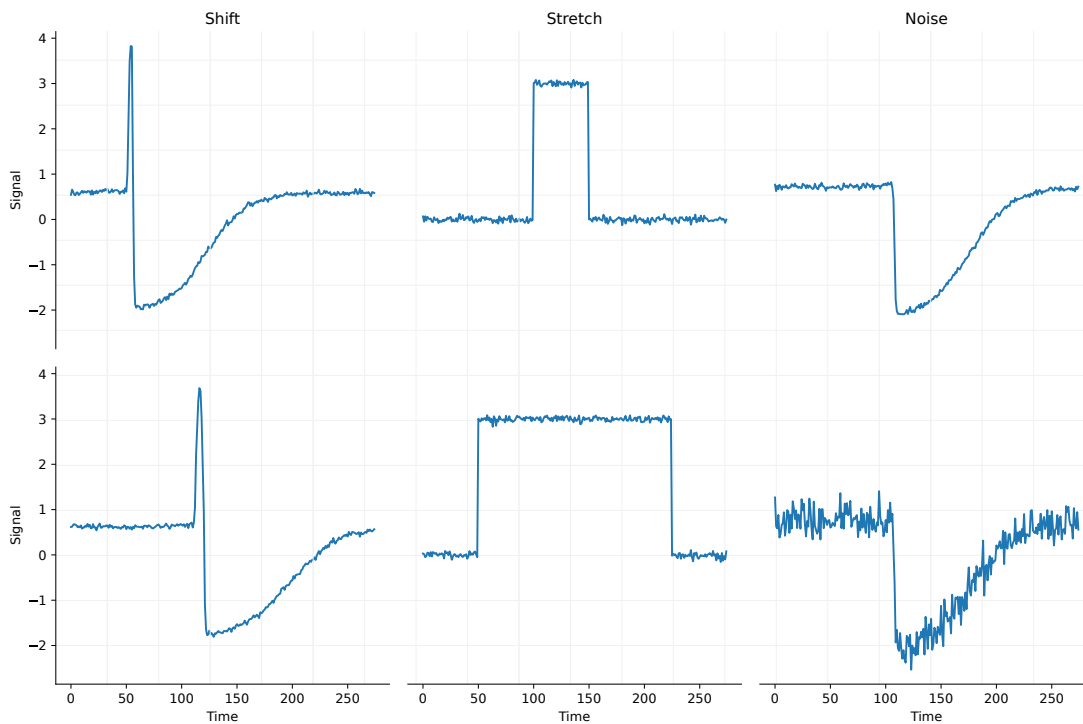


Figure 2.1: Issues with time series clustering in Euclidean space: Examples of time series belonging to the same class with time shifts (left-most column) stretches (central column) or varying noise (right-most column). (Adapted from [103]).

2.1.1 Classical methods

As previously mentioned, unsupervised learning methods in machine learning provide a way to explore and understand the intrinsic structure of data without labeled information or predetermined categories. For time series in particular, special care must be taken to deal with issues such as autocorrelation in the time dimension. In this section, we explore classical approaches (that is, algorithms that do not rely on neural networks) to take care of this issue in time series segmentation, including time-aware feature extraction, specific distance metrics (such as DTW) and HMMs.

2.1.1.1 Classical clustering using time-aware feature extraction

The first approach to explore aims to collapse the time dimension using feature extraction techniques, thus reducing two-dimensional matrices (where dimensions correspond to time and features) to vectors that can be fed to classical dimensionality reduction and clustering algorithms.

This is accomplished by extracting meaningful and representative features that capture the essential patterns and characteristics of the data. Some commonly used feature extraction methods include statistical measures (such as mean, variance, skewness, and kurtosis), frequency domain features (such as Fourier and wavelet transforms), and time-domain features (such as autocorrelation, trend, and seasonality) [107].

Once features are extracted, classical clustering algorithms, such as K-means, hierarchical clustering, or density-based spatial clustering of applications with noise (DBSCAN), can be applied to group similar time series based on their extracted features [52]. As customary with non-time-series data as well, and given the vast number of extracted features in some cases [105] high-dimensional vectors are usually reduced to lower-dimensional manifolds using dimensionality reduction techniques such as Principal Component Analysis (PCA), or Uniform Manifold Approximation and Projection (UMAP) [108, 109]. This helps deal with the so-called “curse of dimensionality”, a phenomenon describing how common distance metrics are not suitable for grouping patterns in high-dimensional spaces [104].

Thus and so, this approach enables the use of well-established clustering techniques on time series data while reducing computational complexity and mitigating challenges associated with the time-dependent nature of the data. However, the success of these kinds of pipelines largely depends on the choice of features and their ability to represent the intrinsic structure of the time series data effectively [110, 111]. Moreover, for large datasets feature extraction itself can become cumbersome in terms of computational complexity, deeming other methods usually more suitable [111].

The standard pipeline included in the python package tsfresh (Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests) is a good example of this kind of approaches in practice. It applies 63 time series characterization methods, which (using default parameters) output a series of 794 features per dimension. Furthermore, it offers domain specific subsets of features, and task-specific feature selection pipelines such

as statistical testing for supervised learning, and variance-based methods for clustering [105] (Figure 2.2).

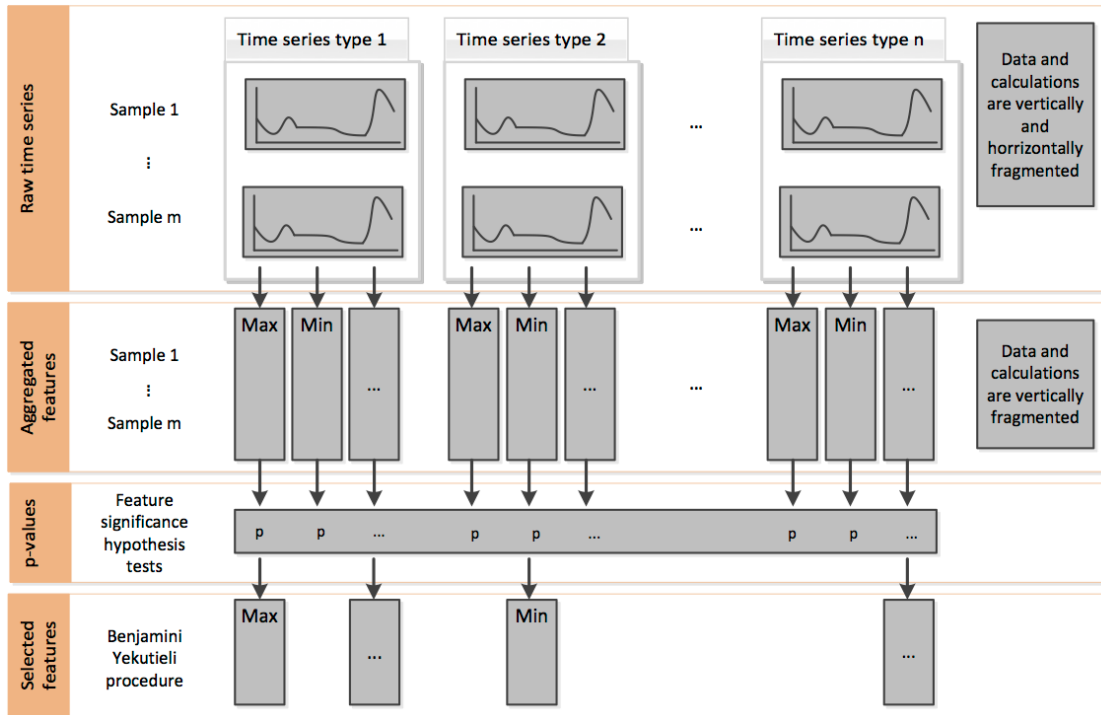


Figure 2.2: An overview of the tsfresh feature extraction pipeline: Starting with the analysis of time series, the algorithm uses established feature mappings and includes extra meta-information features. When focused on supervised learning tasks, each feature is individually assessed to gauge its predictive importance for the target in question. The process involves statistical testing and p -value adjusting for multiple comparisons, which helps decide which features should be retained. In contrast, for unsupervised learning tasks, the algorithm applies variance-based filters on the features. (Adapted from [105]).

2.1.1.2 Dynamic Time Warping and temporal K-Means

While the approach described so far aimed to adapt time series to work with already established clustering methods, this section deals with the arguably opposite approach: by defining suitable distance metrics, algorithms can be adapted to work with time series data without the need for custom feature extraction pipelines. This eliminates the need for domain knowledge or extensive experimentation to identify the most representative features, which can be a time-consuming process.

Along these lines, Dynamic Time Warping (DTW) is a powerful technique for measuring the similarity between two time series, allowing for non-linear alignments that can account for variations in the timing and speed of patterns within the data. DTW works

by dynamically aligning the sequences and calculating the optimal distance between them, taking into account potential time shifts and warping of the series (Figure 2.3).

Formally, let us consider two time series x and x' , where all elements x_i and x'_j lie in the same d -dimensional space. The exact points in time where patterns occur are ignored, and just the ordering of the sequence matters.

The algorithm then searches for the alignment that minimizes the Euclidean distance between two time series:

$$\text{DTW}_q(x, x') = \min_{\pi \in \mathcal{A}(x, x')} \left(\sum_{(i, j) \in \pi} d(x_i, x'_j)^q \right)^{\frac{1}{q}} \quad (2.3)$$

where π is an alignment path of length K (a sequence of K index pairs), and $\mathcal{A}(x, x')$ is the set of all admissible paths (where indices are monotonically increasing, and start and end of both time series match) [106, 112, 113, 114]. All in all, the algorithm returns distances that are always positive ($\text{DTW}_q(x, x') \geq 0$) for any time series x and x' . Moreover, the DTW distance between any time series and itself is always zero ($\text{DTW}_q(x, x) = 0$).

When applied to clustering, the most commonly used algorithm is known as temporal K-means, which is an adaptation of the K-means algorithm that (among other modifications, such as DTW Barycenter Averaging) uses DTW as a distance metric instead of its Euclidean counterpart [115]. Interestingly, when compared to the previously described approaches, DTW-based methods tend to be less sensitive to noise in the data. However, they are also less prone to detecting changes in amplitude, as they focus on the overall shape of the time series. Furthermore, DTW-based methods can be computationally expensive, particularly for large datasets or long time series, as they require calculating pairwise distances between all data points in the sequences (although this can be mitigated by applying bounding techniques to prune the search space [116]). All in all, DTW remains a popular choice for time series clustering due to its flexibility and ability to capture complex relationships within the data.

Finally, it is worth mentioning that the approaches described so far aim in principle to group time series rather than to find segments within them. This can be solved, however, by extracting sliding windows within a given time series, as will be explored in more detail later in this and the following chapter [78, 80]. The next section deals with Hidden Markov Models (HMMs), a set of approaches which can, among other things, handle segmentation tasks directly, without the need for sliding window extraction.

2.1.1.3 Hidden Markov Models

Hidden Markov Models fall in the category of probabilistic graphical models (PGMs), which model sets of (observed or latent) variables as a joint probability distribution in a way that aims to encode conditional independence assumptions using a graph structure [104].

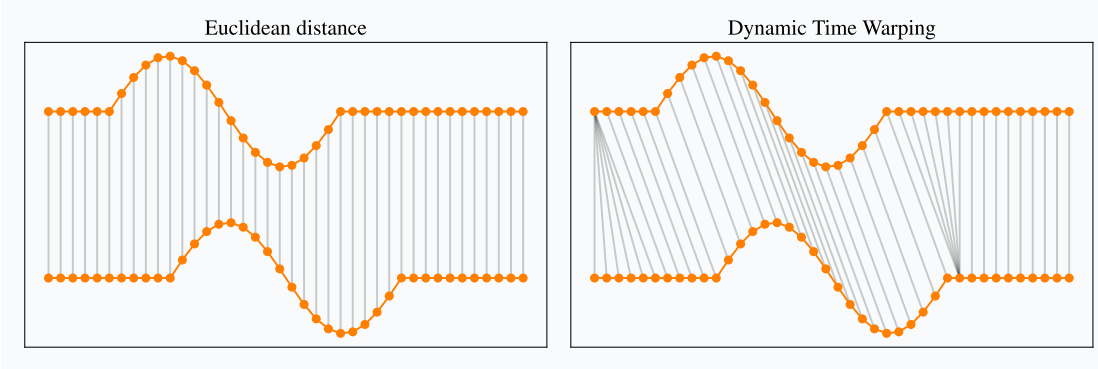


Figure 2.3: Comparison between DTW and Euclidean distance: Comparison between two time series using either Euclidean distance (shown on the left) or Dynamic Time Warping (DTW, shown on the right), the latter being a classic example of an alignment-based metric. In both scenarios, the computed similarity is the cumulative distance between corresponding features (indicated by the gray lines). It is noteworthy how DTW aligns distinctive patterns in the time series, which results in a method that is likely to provide a more reliable evaluation of similarity for time series than the Euclidean distance approach, which aligns timestamps irrespective of their feature values. (Adapted from [106]).

The fundamental concept in PGMs is that every node in the graph symbolizes a random variable, and each edge signifies a direct dependency. Furthermore, the absence of an edge indicates conditional independence between two variables. In the case of a directed acyclic graph (DAG, often referred to as a Bayesian network), the nodes can be arranged in topological order (with parents preceding their children) and connected in such a way that each node is conditionally independent of its predecessors, given its parent nodes:

$$Y_i \perp Y_{\text{pred}_i \setminus \text{pa}_i} \mid Y_{\text{pa}_i} \quad (2.4)$$

where pa_i are the parents of node i , and pred_i are the predecessors of node i in the given ordering. The joint distribution can thus be represented as follows:

$$p(Y_1 : N_G) = \prod_{i=1}^{N_G} p(Y_i | Y_{\text{pa}_i}) \quad (2.5)$$

where N_G is the number of nodes in the graph.

Along these lines, an HMM is a graphical model that models observations across time (y) as coming from a set of latent discrete states (z). Once trained, the model will

2 State of the Art

assign a state to each time point in the series, which depends both on the observations available for that time point, and on the state assigned to the time point that came just before (Figure 2.4, **a**). This last statement corresponds to the Markov assumption, which applies to the latent (hidden) variables, hence the name of the models. Formally, the joint probability of the model can be represented as:

$$p(y_1 : T, z_1 : T) = p(z_1) \prod_{t=2}^T p(z_t | z_{t-1}) \prod_{t=1}^T p(y_t | z_t) \quad (2.6)$$

where z_t are the hidden variables, and y_t are the observations (outputs) at time t . In practice, training such models requires learning probability distributions describing each of the states (called emission distributions), as well as a transition matrix describing the probability of any given state transition. As an example, a hidden Markov model with two states could be applied to represent the rolls of a fair and a loaded dice (respectively) in a casino. By estimating the transition probabilities between the states, a model like this could enable the identification of possible cheating or unfair play given a sequence of dice rolls [104] (Figure 2.4, **b**).

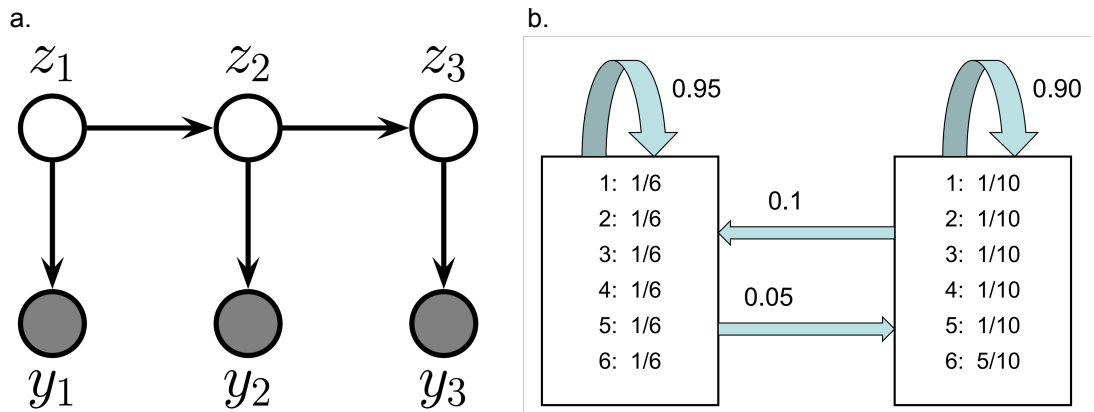


Figure 2.4: Hidden Markov Models (HMMs): **a.** An HMM represented as a graphical model, where z_t are the hidden variables at time t , and y_t are the observations (outputs). **b.** State probabilities for a fair and a loaded dice (left and right, respectively), alongside transition probabilities between them (light blue arrows). (Adapted from [117]).

As previously introduced, an advantage of these models when compared to the previously discussed approaches is that HMMs segment each time series directly, without the need to discretize the time dimension using sliding windows. Moreover, the parametric nature of the models has strong advantages when it comes to interpretability, as each state is described by probability distributions that can be mapped back to the data. However, these advantages come at the expense of flexibility, as long range dependencies

in the data are lost by definition. While extensions to these models have been developed to mitigate this issue, their correct implementation requires domain expertise and may not be applicable to all scenarios [76, 118].

In strong contrast to this idea, the next section delves into the use of neural networks to model time dependencies, and into the existing deep learning algorithms for time series clustering and segmentation. Interestingly, these models follow the opposite trend as HMMs: while neural networks are universal function approximators that arguably champion flexibility, their interpretation often requires significantly more effort [50].

2.1.2 Deep clustering

Representation learning is a subfield of machine learning that focuses on finding transformations that can automatically discover abstract, meaningful features from raw data [50]. These features can then be used to improve the performance of various tasks such as classification, regression, and clustering. Moreover, deep neural networks (DNNs) are capable of capturing complex patterns and hierarchical structures in the data, making them extremely useful for learning meaningful representations.

Thus and so, and in contrast to the methods presented so far, deep learning approaches for clustering typically involve learning a representation of the data and performing clustering on the result of this transformation rather than on the raw data, either jointly or in a post-hoc fashion [103]. This representation is obtained by encoding the data with a DNN (referred to as an encoder) capable of acting as a non-linear mapping function $f_{\Theta} : X \rightarrow Z$, where Θ represents all learnable parameters. These models can thus learn Z as a representation of X , which is called the latent (or hidden) space. The clustering task then involves partitioning the set Z , such that:

$$Z = \{z_1, \dots, z_N\} = \{f_{\Theta}(x_1), \dots, f_{\Theta}(x_N)\} \quad (2.7)$$

where the partition is defined over Z (which in turn is a function of X) instead of over X directly as presented in equation 2.1. Moreover, network architectures, the data and its processing, as well as training schemes used are crucial to a successful representation.

Along these lines, an extremely popular architecture for representation learning has been the deep autoencoder. Its basic architecture consists of two parts: an encoder that maps the input data to a lower-dimensional latent space $f_{\Theta} : X \rightarrow Z$, and a decoder that reconstructs the input data from the latent representation $g_{\Theta} : Z \rightarrow y$ [81]. The objective of an autoencoder is then to minimize the reconstruction error of the input given the output $p(X|y)$ while forcing the data through a series of bottleneck layers that impose constraints in the model, preventing it from learning trivial solutions such as the identity function. Moreover, adaptations such as the Variational Autoencoder [119] enabled its effective use for representation learning and data generation, since the latent space can be interpreted as a probability distribution.

Another popular approach for representation learning is contrastive learning, which works using an encoder only, by optimizing a contrastive loss function that encourages the model to pull together positive pairs (similar instances) and push apart negative pairs (dissimilar instances). This set of approaches has been particularly successful in self-supervised learning scenarios, where large amounts of unlabeled data are leveraged to learn useful representations [120].

Furthermore, and while representation learning is a crucial step of the pipeline, deep clustering goes a step further by aiming to segment the learned manifolds into meaningful subgroups. Either by learning representations that facilitate clustering and segmenting afterward or by training a clustering solution jointly with the representation, these approaches have several advantages. Besides the ability to learn non-linear transformations of the data, which can lead to better cluster separation, they can automatically discover hierarchical structures in the data, and be more robust to noise and irrelevant features due to the hierarchical nature of the learned representations. Moreover, end-to-end models allow for back-propagation of the clustering structure through the encoder, priming the network as a whole to yield representations that have a cluster structure [103].

Before delving into deep clustering itself in the following chapters, the next sections aim to establish a common ground on the building blocks of working with time series and deep neural networks. Three architectural paradigms are presented, including Recurrent Neural Networks (RNNs), Temporal Convolutional Networks (TCNs), and Transformer networks.

2.1.2.1 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are a class of deep learning models specifically designed to handle sequential data, which makes them highly applicable for tasks involving time series analysis [121]. The core feature of an RNN is its ability to maintain a hidden state that captures information from previous time steps, allowing any given model to effectively process and learn from temporal dependencies within the data. This structure enables RNNs to excel in a wide range of applications, such as natural language processing, speech recognition, and, as this section suggests, time series representation in general.

They have the particularity that the sequence is fed step by step to the layer, updating a common hidden state, which serves as a memory of preceding steps. Thus, the layer consists of a recursive function g that takes the current data step x_t and the previous hidden state h_t to generate the new updated hidden state:

$$h_t = g(x_t, h_{t-1}) \tag{2.8}$$

where h_0 is typically initialized with zeros. The original recursive function was defined as:

$$h_t = \tanh(Wx_t + Uh_{t-1} + b) \quad (2.9)$$

where h_t is a vector of size u , with $W \in \mathbb{R}^{u \times d}$ and $U \in \mathbb{R}^{u \times u}$ representing the weights, and $b \in \mathbb{R}^u$ denoting the bias vector learned during training. While useful in many cases, this cell type often results in vanishing gradients, making training difficult and long-range dependency learning hard [50]. Alternative formulations, such as Gated Recurrent Units (GRU) and Long Short-Term Memory (LSTM) cells [122, 123], have been then proposed (Figure 2.5).

In a GRU layer there are three subunits, also called gates, controlling the hidden state update and output called the update gate, the reset gate, and the candidate gate. They are calculated respectively as:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (2.10)$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (2.11)$$

$$\hat{h}_t = \tanh(W_h x_t + U_h (r_t \circ h_{t-1}) + b_h) \quad (2.12)$$

where σ represents the sigmoid function, and \circ denotes the element-wise product. The hidden state is then updated by combining these gates using a specific recursive function:

$$h_t = (1 - z_t) \circ h_{t-1} + z_t \circ \hat{h}_t \quad (2.13)$$

An LSTM unit has in turn more parameters, and it consists of four gates, called the input gate, output gate, forget gate, and candidate memory gate. These subunits are calculated as follows:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2.14)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (2.15)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (2.16)$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (2.17)$$

The memory cell is computed using these gates in yet another recursive function:

$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \quad (2.18)$$

2 State of the Art

The hidden state is then updated as:

$$h_t = o_t \circ \tanh(c_t) \quad (2.19)$$

Once these layers are trained, the hidden state h_T at the end of the sequence is typically considered a learned representation which, in the cases of the autoencoding and contrastive architectures explored in the previous section, can act as inputs to decoder networks or be the target of the contrastive loss, respectively.

Moreover, traditional RNNs process the data sequentially and therefore have no access to future events when processing a given time point. While in many situations this is a desirable property, a successful representation often benefits from bidirectional layers, which involve training two parallel RNN layers with one processing the time steps in the time order (1 to T) and the other processing them backward (T to 1).

All in all, recurrent neural networks are a widely spread architecture for deep sequence processing, including time series. While outperformed by other alternatives in many scenarios, it is worth noting that the recycled parameters across time points result in relative small models in comparison to those that follow, which can have strong advantages when data size is limited.

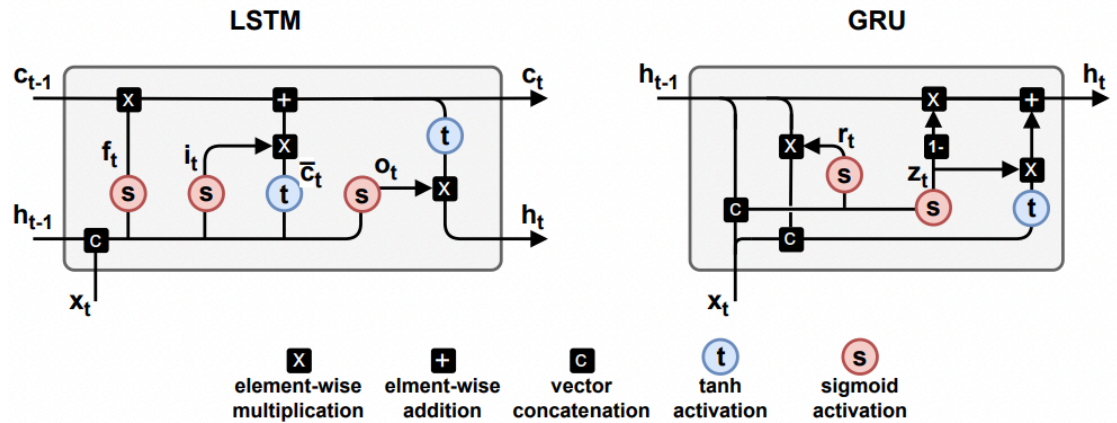


Figure 2.5: Recurrent Neural Networks (RNNs): Schemes representing LSTM and GRU cells. Both show their input x_t and the computation of the new hidden state h_t ; the LSTM on the left also depicts the additional computation of the cell state c_t . (Adapted from [103]).

2.1.2.2 Temporal Convolutional Networks

Temporal Convolutional Networks (TCNs) are another type of neural network architecture that are designed to handle sequence data, but with a distinctly different approach than RNNs. Instead of maintaining a hidden state over time, TCNs leverage a special-

ized form of 1-dimensional convolution, which is applied across the temporal dimension of the input data. A key feature of classical TCNs is the use of causal padding to ensure that future data does not influence the current output, thereby preserving the temporal ordering of the sequence. This contrasts with the bidirectional recurrent layers mentioned above, but can be modified in situations in which looking into the future is not necessarily a problem, such as offline time series segmentation. All in all, TCNs are highly versatile and have been used successfully in a variety of applications, such as audio generation and machine translation [124].

While a simple causal convolution is only able to look back at a history with size linear in the depth of the network, there are several architectural modifications that can be added to solve this issue (Figure 2.6). For starters, the use of dilated convolutions enables the model to have an exponentially large receptive field [125]. Formally, for a 1-D sequence input $X \in R_n$ and a filter $f : \{0, \dots, k - 1\} \rightarrow R$, the dilated convolution operation F on an element s of the sequence is defined as

$$F(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \quad (2.20)$$

where k is the filter size, d is the dilation factor, and $s - d \cdot i$ refers to previous time points. The dilation operation is then equivalent to introducing a fixed step between every two adjacent filters. When $d = 1$, a dilated convolution is equivalent to a regular convolution. Using larger dilations can thus effectively expand the receptive field of the layer. As a corollary, there are two ways to increase the receptive field of the TCN: choosing larger filter sizes k and increasing the dilation factor d , where the effective history of one such layer is $(k - 1)d$.

Another popular modification to this architecture is the addition of residual blocks [64]. These consist of branches in the network leading out to a series of transformations F , whose outputs are added to the input x of the block: $y = \text{Activation}(x + F(x))$. This effectively allows layers to learn modifications to the identity mapping rather than the entire transformation, which has repeatedly been shown to benefit very deep networks. Within a given residual block, the TCN architecture has two layers of dilated causal convolutions and a non-linearity. Weight normalization and is typically added to the convolutional filters, and dropout is introduced for regularization purposes [124].

While a powerful framework that is perfectly suitable for the task presented in this thesis, RNNs and TCNs are increasingly being outperformed by models based on self-attention, such as Transformers. In the next section, we will explore what these are and the advantages they offer in terms of receptive field and interpretability, and the disadvantages they may pose when it comes to computing power and data requirements.

2 State of the Art

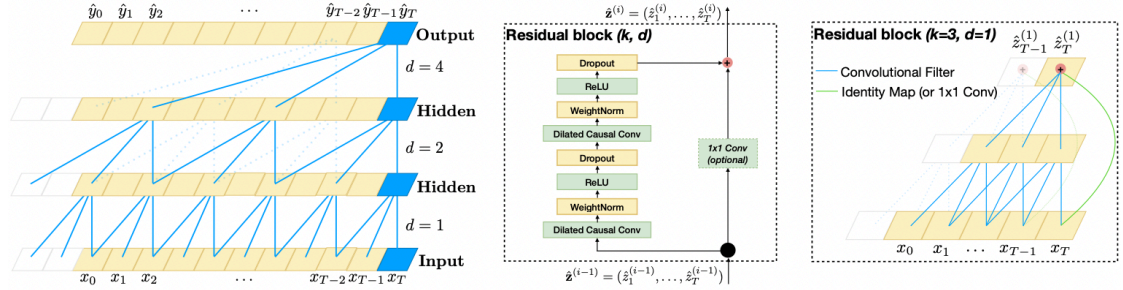


Figure 2.6: Architectural elements of a Temporal Convolutional Network (TCN): **Left:** A causal convolution with dilation factors of $d = 1, 2, 4$, and a filter size of $k = 3$ is shown. This receptive field can cover all input sequence values. **Center:** This is a Temporal Convolutional Network (TCN) residual block. When the residual input and output have different dimensions, a 1×1 convolution is introduced. **Right:** This illustrates a residual connection in a TCN. The blue lines represent filters in the residual function, and the green lines symbolize identity mappings. (Adapted from [124]).

2.1.2.3 Transformer Networks

The Transformer architecture, introduced by Vaswani et al. in the paper “Attention is All You Need” [126] is a novel approach to sequence-to-sequence tasks that significantly improves over traditional Recurrent Neural Networks (RNNs) and (Temporal) Convolutional Neural Networks (TCNs), especially for scenarios where large amounts of data are available.

Transformers are based on the concept of self-attention mechanisms, which allows them to process input sequences in parallel rather than sequentially. This means that sequential information is not directly available to them, and explicit positional encodings are needed to retain order information. The input is thus embedded together with these encodings, and the result is passed through multiple stacked layers of multi-head self-attention and position-wise feed-forward networks. Each layer has residual connections and is followed by layer normalization.

The so called self-attention mechanism works by computing a weighted sum over the input elements and calculating attention scores using a scaled dot-product attention (Figure 2.7, left):

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V \quad (2.21)$$

Here, Q , K , and V are the query, key, and value matrices, respectively, and d_k is the key dimension. This mechanism has several advantages over traditional sequence-aware deep learning methods, since the model can effectively model global dependencies by weighting all time points simultaneously. Moreover, the retrieved attention scores can

2.2 Segmenting behavior: exploring available approaches

aid with model interpretability, since they provided a direct measure of features that the model considers important for a particular task.

Furthermore, Transformer networks use a mechanism called Multi-head attention, which applies self-attention multiple times in parallel, concatenating the outputs, and linearly transforming the result (Figure 2.7, right):

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (2.22)$$

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad (2.23)$$

Here, W^Q , W^K , W^V , and W^O are learnable weight matrices.

The Transformer architecture consists of an encoder-decoder structure. The encoder is composed of a stack of identical layers with multi-head self-attention and position-wise feed-forward networks. The decoder has a similar structure but includes an additional multi-head attention mechanism that attends to the encoder’s output.

The final output of the Transformer is then produced by a linear layer followed by a *softmax* layer, yielding a series of probabilities over a set of tokens.

Transformers have demonstrated exceptional performance in various natural language processing tasks, such as machine translation, text summarization, and question answering. Moreover, they are showing promising results in multi-animal motion tracking models that require complex deidentification of individuals [67].

While efficient due to its highly parallelizable architecture, Transformers are well known for requiring vast amounts of data and computing power to work in practice. While the current thesis includes experiments using models akin to these, data sizes achieved in behavioral experiment setups are arguably too small for the task. Now that several time series processing architectures and clustering methods have been presented, the next sections will delve into deployed algorithms for motion time series segmentation that use some of the discussed approaches.

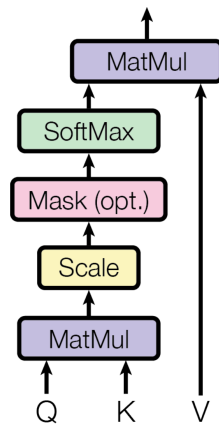
2.2 Segmenting behavior: exploring available approaches

Since the advent of DeepLabCut and SLEAP, a significant number of tools designed to leverage marker-less pose estimation have been introduced. The subsequent sections explore three such tools, all of which provide approaches to behavioral segmentation. These are B-SOiD [78], which utilizes traditional clustering techniques on features extracted over time, MoSeq [76] which relies on Hidden Markov Models, and VAME, which takes advantage of post-hoc clustering on embeddings generated through deep neural networks.

2.2.0.1 B-SOiD: time series clustering using guided representations

B-SOiD was presented in 2021 by Alexander Hsu and Eric Yttri [78]. It works by extracting a series of descriptive kinematic features from the motion tracking time series

Scaled Dot-Product Attention



Multi-Head Attention

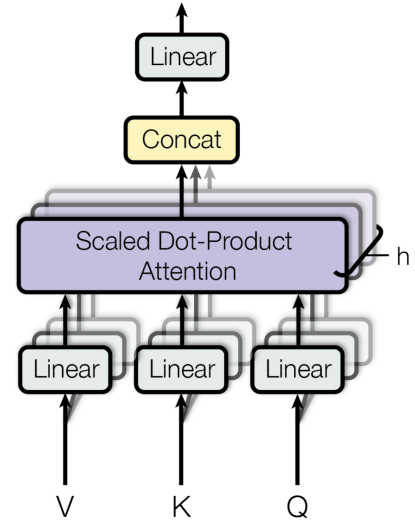


Figure 2.7: Details on the transformer architecture. The **Scaled Dot-Product Attention** operation is depicted on the left, with Queries, Keys, and Values as inputs. The right panel shows the **Multi-Head Attention** mechanism, which parallelizes several attention heads, in a fashion that can be compared to ensemble learning. (Adapted from [126]).

obtained with DeepLabCut or SLEAP, including displacement, angular change, and distances between body parts. These features are then aggregated over a sliding window of 60 ms (30 ms before and 30 ms after the frame of interest). Furthermore, the original data is downsampled to 10 frames per second (FPS), which the authors claim helps increase the signal-to-noise when it comes to distinguishing real movement from label jittering.

Once these features are extracted, their dimensionality is reduced using PCA, with a number of components such that they explain 70% of the variance in the motion tracking data. UMAP is then applied with the same number of components to get a representation that enforces local aggregation, and motion clusters are obtained using HDBSCAN. This algorithm is a particularly good approach when it comes to detecting outliers, since it can remove subthreshold densities [127]. Moreover, it does not require the user to select the number of clusters ad-hoc.

Once the clusters are obtained, B-SOiD trains a multi-class random forest from the original statistics to the cluster labels assigned by HDBSCAN, which is shown to improve generalizability to unseen data. The trained models can then quickly assign data points to clusters and enable further analysis (Figure 2.8).

While a simple approach, B-SOiD is shown to work quite well in many popular scenarios including not only rodents, but also flies and humans, to name a few. Moreover,

2.2 Segmenting behavior: exploring available approaches

it does not require expensive hardware to train in small or medium datasets, and further developments use the obtained clusters as the starting point to train even more generalizable classifiers in an active learning framework [77]. The simplicity of the pipeline has its limitations, however, as the extracted features are assumed to capture most variation in the detected animal’s dynamics. In rodents, this means that direct access to paw movement is needed to achieve good results, which is only achieved with bottom-up videos (where the animals are filmed from below through a glass floor). This is non-standard practice in many labs, since it requires special hardware and it was shown to stress the animals [68]. Furthermore, no explicit information about dynamics is used, which can hurdle the flexibility of the retrieved clusters, and only single animals are supported.

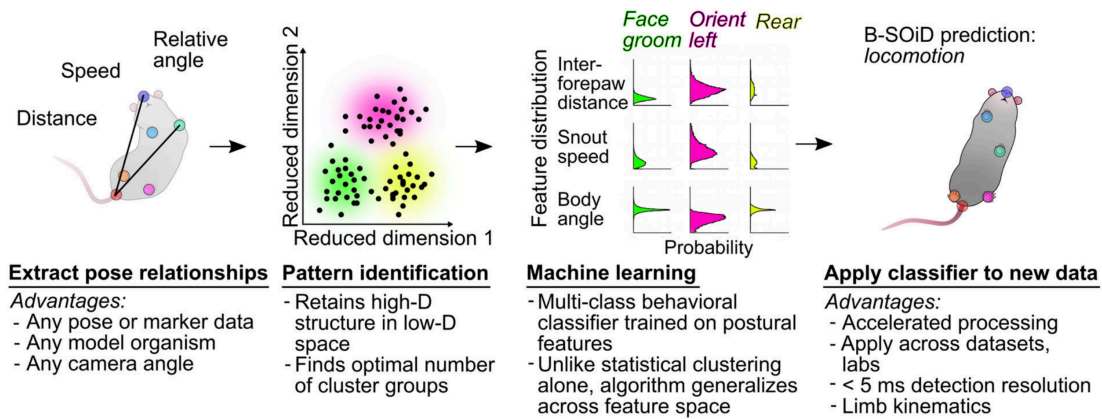


Figure 2.8: Overview of the B-SOiD pipeline. Once the pose relationships that characterize behaviors are extracted, B-SOiD applies a non-linear transformation (UMAP) to preserve high-dimensional postural time-series data in a lower-dimensional space, and HDBSCAN is subsequently used for cluster identification. The spatiotemporal features that have been clustered serve as inputs for training a random forest classifier, which can then be employed to promptly predict behavioral categories in any comparable data set. Once trained, the model will segment any dataset into the same classes. (Adapted from [78]).

2.2.0.2 MoSeq: motion clustering with Autoregressive Hidden Markov Models

MoSeq (motion sequencing) is a Hidden Markov Model approach for behavioral segmentation introduced by Sandeep Robert Datta and collaborators back in 2019 [128]. To solve the short range dependencies introduced by the Markov assumption, the authors use an Autoregressive Hidden Markov Model (AR-HMM) approach, which relies on an HMM where the observation model (the model that generates the outputs from the hidden states) is an autoregressive model in itself. In other words, the models are designed so that each hidden state has an autoregressive (AR) model associated with it, which means that the current output not only depends on the current hidden state, but also on previous outputs.

2 State of the Art

While the authors originally demonstrated high capabilities of this approach for depth sensing camera setups, the original algorithms were not capable of dealing with keypoint estimation data coming from regular video. In a recent preprint [76], the group behind MoSeq introduced a variant which decouples keypoint motion to actual animal motion and label jitter, which the authors identify as the main culprit of previous versions’ poor performance in these settings (Figure 2.9).

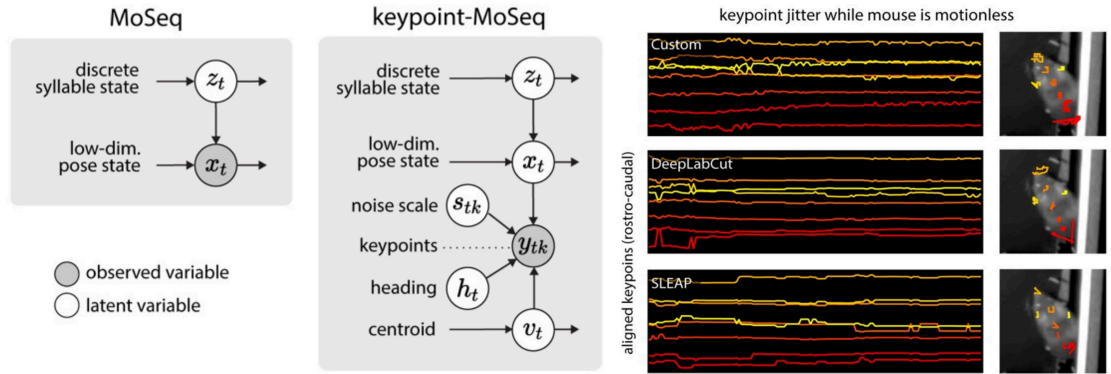


Figure 2.9: MoSeq: motion clustering with Autoregressive Hidden Markov Models. **a:** Graphical models showcasing both MoSeq and a new hierarchical model known as “keypoint-MoSeq” are presented. In both models, a discrete syllable sequence dictates the dynamics of a low-dimensional pose state. The pose state is either determined using PCA (as demonstrated in “MoSeq”, left) or inferred from keypoint observations in relation to the animal’s centroid and heading, as well as a noise scale to account for keypoint detection errors (as demonstrated in “keypoint-MoSeq”, right). **b:** This is an example of keypoint jitter from three distinct keypoint tracking methods during a 5-second interval when the mouse was stationary. The left side shows keypoint trajectories aligned egocentrically, whereas the right side shows the path traced by each keypoint during the interval. (Adapted from [76]).

2.2.0.3 VAME: Variational Animal Motion Embeddings

Finally, Kevin Luxem and colleagues introduced VAME (Variational Animal Motion Embeddings) in 2022 (Figure 2.10). The package, implemented in Python and PyTorch, relies on deep neural networks to embed motion tracking time series. In their pipeline, a variational autoencoder maps the input to an unimodal multivariate Gaussian latent space, and post-hoc clustering is applied to the embeddings using a Hidden Markov Model. Moreover, the architecture has two decoders: one trained to minimize the reconstruction error with the input (reconstruction decoder), and one that aims to predict the next unseen timepoint (prediction decoder), which helps to regularize the obtained embeddings. In their paper [80], the authors show how their models can robustly detect shifts in behavior in rodents with beta amyloidosis. While results are promising, the authors only demonstrate the capabilities of the software using bottom-up videos with access to the paws’ positions at all times.

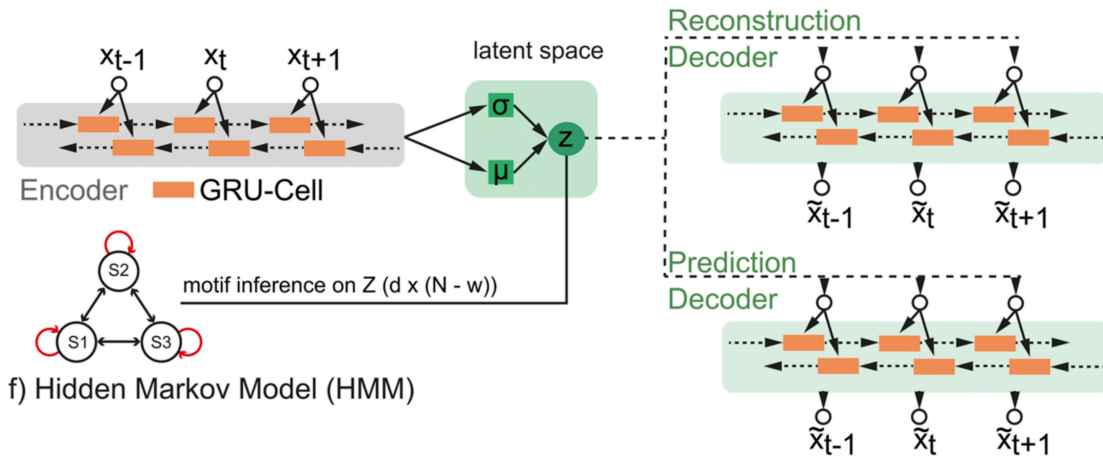


Figure 2.10: VAME: Variational Animal Motion Embeddings. Frames are aligned ego-centrally, and trajectory samples are fed to a recurrent variational autoencoder model. The fully trained model functions like a dynamical system, from which motifs are deduced using a Hidden-Markov-Model. (Adapted from [80]).

2.3 Main contributions of this thesis to the field

All in all, this thesis aims to build on the presented state of the art in three main ways.

2.3.1 Implementation and testing of novel deep clustering algorithms for unsupervised behavioral segmentation

As a first goal, this thesis aims to present new variants of deep clustering algorithms (that is, models that couple deep neural network embeddings and clustering) for multivariate time series data, specifically tailored to work with rodent motion tracking.

2.3.2 Deployment of the developed algorithms to the community

Aside from implementing and testing these developed algorithms, an important goal of this work is to deploy them in a packaged manner for the community to use with their own data. Along these lines the DeepOF (Deep Open Field) package was born, a Python suite with tools for processing, annotation, and deep clustering of rodent motion tracking data. Chapter 4 will delve into the design philosophy of DeepOF, its inner workings, and include a paper published in the Journal of Open Source Software (JOSS), which included detailed code, documentation, and testing pipeline reviews.

2.3.3 Application of the developed algorithms to the characterization of Chronic Social Defeat Stress

As introduced earlier in chapter 1, all developed algorithms were applied to the characterization of Chronic Social Defeat Stress as a case study. Chapter 5 will delve into this, presenting the deployed algorithms in detail, and showcasing the application of DeepOF to the supervised and unsupervised characterization of the provided animal model, as a paper published in Nature Communications.

3 Methods

3 Methods

While most methods are detailed in the papers presented in the following chapters, these sections aim to fill certain gaps and delve into the developed algorithms included in DeepOF that were not applied in the results presented in chapter 5.

3.1 Software architecture and deployment

The software package presented in this thesis, called DeepOF, was developed using a modular approach consisting of three primary modules designed for user interaction (called `deepof.data`, `deepof.post_hoc`, and `deepof.visuals`), and five modules for internal calculations. A comprehensive list and description of all modules follows:

- `deepof.data` - Includes tools for data loading, preprocessing, and pattern extraction
- `deepof.post_hoc` - Includes tools for post-hoc analysis tools for results obtained with annotation pipelines
- `deepof.visuals` - Includes a comprehensive set of visualization functions
- `deepof.utils` - Includes general utilities
- `deepof.models` - Contains code for the deep clustering model architectures
- `deepof.hypermodels` - Contains hypermodels for hyperparameter tuning of the deep clustering models
- `deepof.annotation_utils` - Contains utilities for the supervised annotation pipeline
- `deepof.model_utils` - Contains utilities for the unsupervised pipeline (including model training and evaluation)

Furthermore, DeepOF also features a suite of automated tests with continuous integration (CI) to ensure the proper functioning of all deployed code. These were implemented using the hypothesis package for Python, a suite that enables property-based testing, where synthetic examples are created on the fly to test all functions while following a set of defined constraints [129]. Test coverage is reported automatically, and computed using the coverage package for Python [130].

Documentation was written and deployed using read-the-docs [131], and automatic API pages were created using `autodoc`. Contributing guidelines and a code of conduct are also included.

3.2 Data loading and input

DeepOF takes a set of two files per experiment that was carried out, including a video (various standard formats are accepted) and a table containing the tracking output

generated with DeepLabCut (which can be in either CSV or HDF). Upon importing the package, the project is initialized as an instance of the `Project` class available in `deepof.data`, and executed using the `.create()` method, which applies all parameters and processing and stores the results in an instance of the `Coordinates` class, also in `deepof.data`. From here, many sets of features can be extracted as instances of the `TableDict` class (also in `deepof.data`), and the supervised and unsupervised annotation pipelines can be executed (see full documentation for more details).

3.3 Time series processing

All relevant details on time series processing can be retrieved from the publication “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”, included as part of chapter 5.

3.4 Supervised annotation of pre-defined traits

All relevant details on the supervised annotation of pre-defined traits can be retrieved from the publication “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”, included as part of chapter 5.

3.5 Unsupervised annotation: exploring the behavioral space

As previously mentioned, DeepOF includes a pipeline for motion tracking time series segmentation, which retrieves behaviors that are expressed consistently throughout the filmed animal experiments. This pipeline provides a series of architectures and data input objects to choose from according to the nature of the captured video. To start with, let us explore how the input for time series segmentation can be represented.

3.5.1 Matrix input/output representations

The simplest way to arrange time series as input for the unsupervised pipeline is the same used by the models from the literature presented in chapter 2. That is, as matrices of features over time, where different attributes are assumed to be independent. Moreover, sliding windows are cross-correlated with these time series, to end with a three-dimensional tensor representation where dimensions correspond to sliding window instances, time within each window, and features, respectively (Figure 3.1, left).

3.5.2 Graph input/output representations

While sufficient in many cases (as in the packages mentioned in chapter 2, where paws are accessible when experimenters are filming from below), matrix representations assume that features are spatially independent, which is not the case. To examine the possibility

3 Methods

of including spatio-temporal relationships between the features, we included in DeepOF the option to represent tracking data as dynamic graphs (Figure 3.1, right). While connectivity in these graphs (whose adjacency matrix links body parts that are spatially adjacent) remains static, features are organized as node and edge attributes that vary across time. Moreover, this enables the program to incorporate more features naturally: this way, not only coordinates but also velocities are included as node attributes, and distances between pairs of body parts are used to annotate edges. Furthermore, this representation can be expanded to accommodate multiple animals, where separate graph representations for each animal are connected via nose-to-nose, nose-to-tail, and tail-to-tail edges, enabling the models to include relative distances between animals. When this is the case, an L_1 penalization over the node embeddings controls the influence social interactions should have in the results, over the posture of individual animals. When using graph representations, inputs to the segmentation models are three fold, and include:

- A four dimensional tensor with **node attributes**, with dimensions corresponding to sliding window instances, time within each window, nodes in the graph, and features in each node.
- A four dimensional tensor with **edge attributes**, with dimensions corresponding to sliding window instances, time within each window, edges in the graph, and features in each node.
- the **adjacency matrix** of the graph to embed, which remains static throughout time.

3.6 Unsupervised annotation: deep clustering models

As part of the first goal of this thesis, as defined in the last section of chapter 2, DeepOF includes three families of deep representation models for time series segmentation. Each of these families (described in the following sections) can be used with matrix or graph input representations. Moreover, their encoder (and decoder, when applicable) structures can be selected from a set of recurrent, TCN, and transformer-based architectures.

When a graph representation is selected as input, these temporal blocks are coupled with graph neural network (GNN) spatial blocks capable of embedding both node and edge attributes [132], building what is known as spatio-temporal graph neural network (ST-GNN) structures [133, 134]. This gives DeepOF flexibility to adapt to different data scenarios and hardware systems, as will be discussed in chapter 6. The next few sections introduce the three families of deep representation models included in DeepOF, which are based in Variational Deep Embeddings (VaDE), Vector Quantization Variational Autoencoders (VQVAE), and self-supervised contrastive learning architectures, respectively. Schematic representations of all three can be found in Figure 3.2.

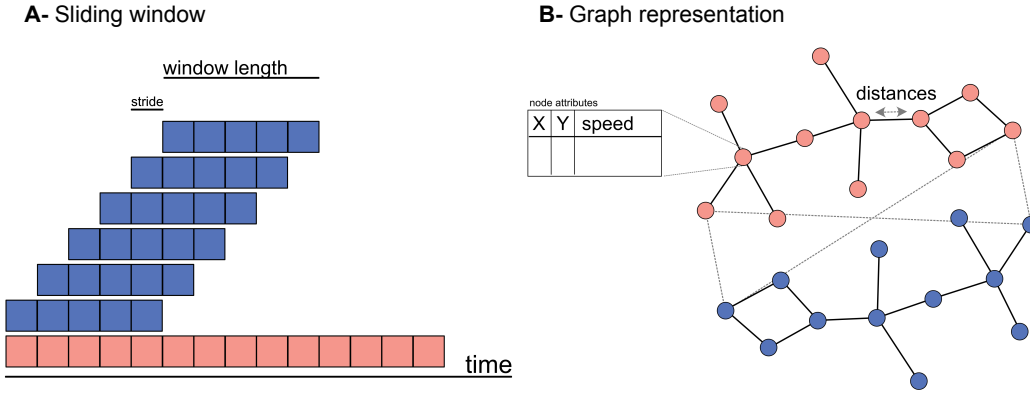


Figure 3.1: Time series input representation for deep clustering of motion tracking data: **A:** Prior to segmentation model training, time series are split using a sliding window approach. The length and stride of the windows are hyperparameters that the user can modify, and default to the frame rate of the videos (so that each window includes one second of motion data) and one, respectively. **B:** To leverage spatial correlations between the features, and allow for the natural inclusion of features other than coordinates, input time series can be represented as dynamic graphs. Here, connectivity remains static, while node and edge attributes vary through time. Moreover, this representation paves the way to include multiple animals in a single model, since edges between body parts of different animals can be incorporated.

3.6.1 Variational Deep Embeddings (VaDE)

The first segmentation architecture available in DeepOF is based on Variational Deep Embeddings (VaDE) [135, 136], a deep clustering algorithm that consists of an encoder-decoder architecture similar to that of a Variational Autoencoder (VAE) [119]. Here, an encoder neural network maps the input X to a latent vector z , and a decoder architecture maps such vector to the output y . As the traditional VAE, the model is trained to minimize the evidence lower bound (ELBO) which is a composite loss function that aims to minimize both the reconstruction error given the input, and the Kullback-Leibler (KL) divergence between the latent vectors and a prior distribution. Unlike the traditional VAE, however, VaDE architectures map the input vectors to a mixture of (in this case Gaussian) distributions, with each component representing a given cluster. Formally, the training process aims to minimize the equation:

$$L_{\text{ELBO}}(x) = \mathbb{E}_{q(z,c|x)}[\log p(x|z)] - D_{\text{KL}}(q(z, c|x) || p(z, c)) \quad (3.1)$$

where the first term corresponds to the reconstruction loss, which encourages the latent space (z) to represent the data (x) well over a set of clusters (c). The second term is the aforementioned KL divergence (D_{KL}) between a Gaussian mixture prior ($p(z, c)$) and the

3 Methods

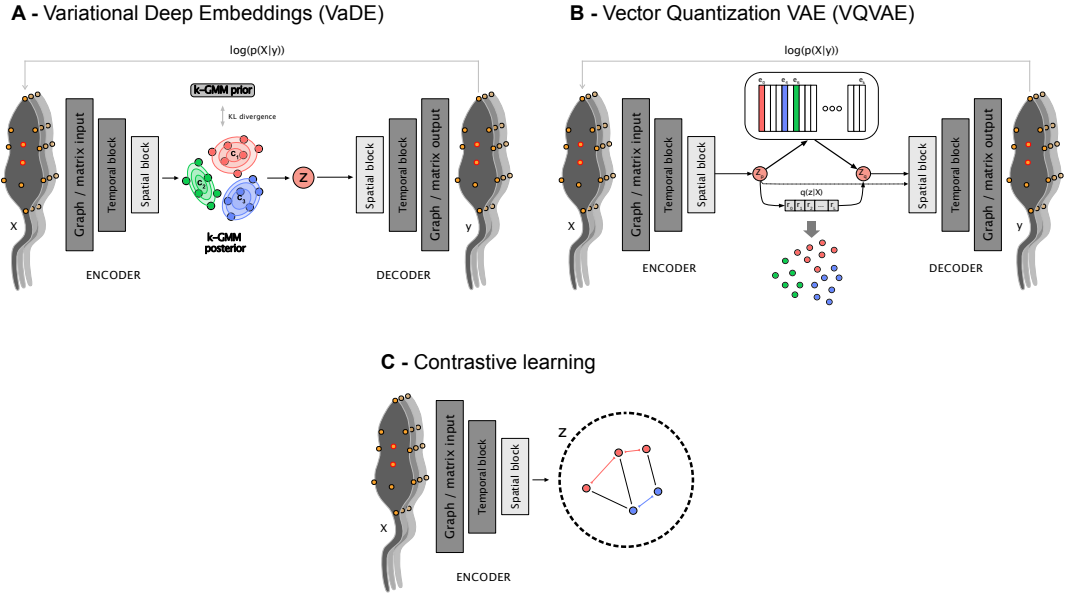


Figure 3.2: Deep clustering architectures implemented within DeepOF: **A:** Variational Deep Embeddings (VaDE): the model follows an encoder-decoder architecture similar to a variational autoencoder, with the key difference that both latent space and prior distribution are mixtures of multivariate Gaussians instead of unimodal distributions. This not only imposes a clustering structure in the latent space Z , but also allows to directly extract soft cluster assignments at inference time, as the normalized likelihood under each component of the mixture. **B:** Vector-Quantization Variational Autoencoders (VQVAE): similar to VaDE, the model follows an autoencoder-like architecture. Instead of having a probability distribution as a prior, there is a discrete codebook being maintained in parallel to the model, whose columns represent cluster centroids on the latent space. At training time, the closest entry in the codebook to the current embedding vector is selected and passed through the decoder instead of the vector itself. The model is trained to reconstruct the input and to minimize the Euclidean distance between each latent vector z and its closest codebook entry. At prediction time, clusters are assigned to the closest codebook entry for the embedding obtained for each input instance. **C:** Contrastive learning: this architecture consists only of an encoder that maps each input instance to a latent space Z . A contrastive loss pulls together similar embeddings (represented with colored arrows between points of the same color, and pushes apart dissimilar embeddings (represented with black arrows between points of different colors). The definition of what *similar* and *dissimilar* mean in this context depends on the loss function of choice. This is the only provided architecture that requires post-hoc clustering of the latent space.

variational posterior for each cluster ($q(z, c|x)$). This serves the purpose of regularizing the embeddings to also follow a Gaussian mixture distribution, where each component is associated with a particular cluster. A schematic overview of the model can be found in Figure 3.2A. Importantly, this loss function imposes a clustering structure directly within

3.6 Unsupervised annotation: deep clustering models

the latent space, eliminating the need for post-hoc clustering of embeddings required by other existing tools. This end-to-end approach offers several benefits, with the primary advantage being that the clustering structure back-propagates to the encoder during training.

After the models are trained, cluster assignments are obtained as the `argmax` of the posterior distribution given the data, as outlined in equation 3.2:

$$q(c|x) = p(c|z) \equiv \frac{p(z)p(z|c)}{\sum_{c'=1}^K p(c')p(z|c')} \quad (3.2)$$

where $c' \in (1, K)$ is an iterator over all clusters in the model. In practice, this unsupervised pipeline can retrieve consistent patterns of animal motion in a flexible, non-linear, and fully unsupervised way. Moreover, training is stable for all encoder-decoder architectures and input structures included in the package. This makes it the ideal default for the unsupervised segmentation pipelines. All results presented in chapter 5 use this architecture.

3.6.2 Vector Quantization Variational Autoencoders (VQVAE)

The second implemented segmentation model to visit is an adapted version of the vector quantization variational autoencoder architecture (VQVAE) [137]. This also follows an encoder-decoder architecture, minimizes the mean squared error reconstruction loss between input and output, and enforces a clustering structure in the latent space in an end-to-end fashion. The main difference with VaDE is that this clustering structure is approached using vector quantization, which enforces a discrete latent space instead of a continuous probability distribution [137]. In practice, the input X is passed through a sequence-aware encoder onto an embedding vector z_p , which is compared to the columns of a separately maintained codebook (represented as a matrix whose column vectors are cluster centroids). The closest codebook column vector (z_q) is then selected (Eq. 3.4) and passed on to the decoder instead of z_p itself:

$$q(z = k|x) = \begin{cases} 1 & \text{for } k = \operatorname{argmin}_j \|z_p(x) - e_j\|_2, \\ 0 & \text{otherwise.} \end{cases} \quad (3.3)$$

$$z_q(x) = e_k, \text{ where } k = \operatorname{argmin}_j \|z_p(x) - e_j\|_2 \quad (3.4)$$

where e_j is a given column of the codebook. The model is then trained to maximize the conditional log-likelihood of the data, $\log(p(X|z_q(x)))$, and minimize the Euclidean distance between z_p and z_q (often referred to as *commitment loss*).

Moreover, a key aspect of the VQVAE setting is that the described lookup operation is non-differentiable, which prevents gradients from flowing through the encoder during

3 Methods

backpropagation, preventing proper training. In the original paper, this problem is overcome by copying the gradients through the lookup (from z_p to z_q). Later work, however, suggested that while such an approximation works well in the image compression setting originally presented, it is not ideal for clustering since the required number of codes is much smaller, which increases the average distance between z_p and z_q during training, making the gradients less informative and leading to a suboptimal encoder [138]. To mitigate this issue, we followed existing approaches and added a second reconstruction loss, which connects z_p with the decoder, bypassing the lookup operation and enabling gradient flow. A scheduler decreases the weight (α) of the loss assigned to this term as training progresses, once the average distances between z_p and z_q are close. The complete loss function is thus defined as:

$$L = \log p(x|z_q(x)) + \alpha \log p(x|z_p(x)) + \beta \|z_p(x) - z_q(x)\|_2^2 \quad (3.5)$$

Once the models are trained, cluster assignments can be obtained using the same lookup operation described in equation 3.4. Moreover, soft counts can be obtained using the fuzzy-c means approach, where confidence is inversely proportional to the distance to the closest centroid [139].

In practice, this model is faster to train than VaDE when all other parameters are left equal, and it is still end-to-end. While this may make it preferable in some hardware-constrained situations, training was shown to be unstable when coupled with graph inputs, making it a poorer overall default choice, especially in top-down video settings. All results presented in the original DeepOF preprint [75], posted in bioRxiv, make use of this architecture.

3.6.3 Contrastive representation learning (CRL)

A third representation and segmentation pipeline is included in DeepOF as an option that, although not end-to-end, I believe deserves to be mentioned. Contrastive learning is a set of representation learning approaches that fall into what the literature has called *self-supervised learning*. In contrast to the two models presented above, which are generative approaches to representation learning (since they directly model the data distribution, and their decoders can explicitly be used to generate data), *self-supervised* approaches use discriminative models instead. This eliminates the need for a potentially wasteful decoder, making learning more efficient when representations are the only goal [120].

Contrastive representation learning in particular works by applying a loss function directly to the latent space, and following a simple principle: *similar* samples (denoted as **positive pairs**) should be pulled together, whereas *dissimilar* samples (**negative pairs**) should be pulled apart. The trick lies in defining what similarity means in this case: given a sample, these approaches sample a positive pair from a positive distribution ($x^+ \sim p^+(\cdot|x)$), and a negative pair from a negative distribution ($x^- \sim p^-(\cdot|x)$). In

3.6 Unsupervised annotation: deep clustering models

DeepOF, positive and negative sampling are based on recently introduced time-series change-point detection contrastive models [140], where samples closer in time have a higher probability of being called a positive pair, and vice versa. Once positive and negative pairs are defined, the default algorithm applies the *InfoNCE* (Noise Contrastive Estimation) loss [141], which maximizes the mutual information between consecutive time windows. Thus, a single positive pair of time adjacent intervals (h_i, f_i , where h_i is called the history window, and f_i the future window), and a set of $K - 1$ negative pairs where the intervals h_i and f_j are well separated in time across the sequence, can be used to calculate the normalized similarity ρ_i across all pairs:

$$\rho_i = \frac{\exp(\text{Sim}(h_i, f_i)/\tau)}{\sum_{j=1}^K \exp(\text{Sim}(h_i, f_j)/\tau)} \quad (3.6)$$

where τ is a scaling parameter and Sim is the cosine similarity between each pair of data embeddings [140]. The final loss is then computed by applying the binary cross-entropy function over the similarities of all pairs:

$$\mathcal{L} = - \sum_{i,j} y_{ij} \log(\rho_i) + (1 - y_{ij}) \log(1 - \rho_i) \quad (3.7)$$

This model has several advantages over the previous two when it comes to learning representations. For starters, it has substantially fewer parameters since it lacks a decoder, which can make training substantially faster. Moreover, the success of contrastive representation learning so far has been linked to improved abstraction (the extraction of concepts that are invariant to local or small changes in the data) and disentanglement (with uncorrelated latent dimensions representing qualitatively different concepts) of representations in many scenarios [120]. However, no noticeable improvements in this regard were obtained so far in DeepOF. Moreover, and as mentioned at the beginning of this section, the provided contrastive models do not enable end-to-end clustering, with cluster assignments needing to be obtained in a post-hoc fashion. DeepOF does this by fitting a Gaussian HMM to the trained latent space [142]. While it does not escape my attention that HMM parameters could be learned jointly with the neural network in theory, thus making the model truly end-to-end, all attempts so far resulted in unstable training. Further efforts in this direction for future releases of the package are not discarded.

3.6.4 Semi-supervised post-hoc reclustering

While both VaDE and VQVAE models offer end-to-end clustering, they do so by grouping similar sliding window instances on a shuffled dataset, without a clear explicit idea of how the sliding windows are ordered across time. While significantly overlapping

3 Methods

windows mitigate this issue, in practice there are sections in the experiments where low-confidence cluster assignments often switch between states, requiring the user to discard them for further analysis. To avoid this problem altogether, DeepOF can train a semi-supervised HMM to the latent space [142], where prior probabilities are assigned to each sample as the VaDE / VQVAE soft counts, and posteriors are obtained by fitting a Gaussian HMM. The overall effect leads to cleaner cluster assignments across time, and longer average times on each cluster.

3.7 Characterization of Chronic Social Defeat Stress (CSDS)

Details on animal handling, CSDS protocols, behavioral testing, datasets used, and experimental design can be retrieved from the publication “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”, included as part of chapter 5.

3.8 DeepOF in practice and post-hoc analysis of annotation results

All relevant details on the post-hoc analysis of both supervised and unsupervised annotation results can be retrieved from the publication “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”, included as part of chapter 5.

3.9 Statistics

Statistical analyses and graphs were made in R (v 4.1.1), python (v 3.9.13), and DeepOF (v0.4.6). Details on tests and assumptions when comparing samples, as well as multiple testing corrections and notation, can be retrieved from the publication “**Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress**”, included as part of chapter 5.

4 DeepOF: a Python package for pattern recognition in mice motion tracking data

4.1 Overview

As previously stated, the first two main goals of this thesis are to implement and deploy tools for deep clustering of motion tracking time series. In this context, DeepOF (deep Open Field) is a Python package that implements tools for loading, processing, and analyzing motion-tracking data. In particular, it provides two analysis pipelines for users to explore: a supervised pipeline, which aims to extract a set of pre-defined patterns from tracked animal trajectories, and an unsupervised pipeline, which applies state-of-the-art deep clustering to segment behavior over time. Moreover, the tool also includes a set of functions to explore the output of these analyses, including pattern expression enrichment and dynamics across experimental conditions, fitting normative models, exploring global shifts in behavior across time, and more. The current chapter includes a paper published in the *Journal of Open Source Software* (JOSS), accepted for publication after a peer-review process that tested both the content of the paper and the proper functioning and writing of the deployed code [143]. At the moment of submitting this thesis, the latest stable version of the package is 0.4.6.

4.2 Package design

DeepOF was implemented following a modular design, with three modules intended for user interaction, called `deepof.data`, `deepof.post_hoc`, and `deepof.visuals`. The first one deals with data loading, preprocessing, and pattern extraction. The second one provides a set of tools for post-hoc analysis of results obtained with the provided annotation pipelines, and the third one includes a plethora of visualization functions. A set of five extra modules contain models and utilities that are not intended for the user to access directly, but are consistently loaded by classes and functions in the public API.






Moreover, DeepOF includes a set of automatic tests deployed with continuous integration (CI), which makes it easier to make sure that all deployed code works as intended. Test coverage is reported automatically as well, to make it easier for maintainers to keep track of what is being tested if the codebase is extended. Extensive documentation is included too (both automatic for the API and manually generated for installation, examples, and tutorials), as well as contributing guidelines and a code of conduct. The language of choice (Python) was selected as the gold standard and most familiar tool for both data science libraries used to implement our models, and the behavioral analysis community as a whole.


All in all, DeepOF was implemented following best practices to make it maintainable and extensible in the future, and we believe it has the potential required to last in the field as a useful and easily accessible tool.

4.3 Contribution to the work

I am to date the sole contributor to the API design and code implementation of the DeepOF package. I also wrote the entirety of the included paper.




DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data

Lucas Miranda ¹, Joeri Bordes ², Benno Pütz ¹, Mathias V Schmidt ², and Bertram Müller-Myhsok ¹✉

¹ Research Group Statistical Genetics, Max Planck Institute of Psychiatry, Munich, Germany ² Research Group Neurobiology of Stress Resilience, Max Planck Institute of Psychiatry, Munich, Germany 
Corresponding author

DOI: [10.21105/joss.05394](https://doi.org/10.21105/joss.05394)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Elizabeth DuPre](#)  

Reviewers:

- [@cellistigs](#)
- [@edeno](#)

Submitted: 12 April 2023

Published: 12 June 2023

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).

Summary

DeepOF (Deep Open Field) is a Python package that provides a suite of tools for analyzing behavior in freely-moving rodents. Specifically, it focuses on postprocessing time-series data extracted from videos using [DeepLabCut](#) ([Mathis et al., 2018](#)). The software encompasses a diverse set of capabilities, such as:

- Loading DeepLabCut data into custom objects and incorporating metadata related to experimental design.
- Processing data, including smoothing, imputation, and feature extraction.
- Annotating behavioral motifs in a supervised manner, such as recognizing huddling and climbing, and detecting fundamental social interactions between animals.
- Embedding motion tracking data in an unsupervised manner using neural network models, which also facilitate end-to-end deep clustering.
- Conducting post-hoc analysis of results and visualization to compare patterns across animals under different experimental conditions.

The package is designed to work with various types of DeepLabCut input (single and multi-animal projects), includes comprehensive documentation, and offers interactive tutorials. Although many of its primary functionalities (particularly the supervised annotation pipeline) were developed with top-down mice videos in mind, tagged with a specific set of labels, most essential functions operate without constraints. As demonstrated in the accompanying scientific application paper ([Bordes et al., 2022](#)), DeepOF has the potential to enable systematic and thorough behavioral assessments in a wide range of preclinical research settings.

Statement of need

The field of behavioral research has experienced significant advancements in recent years, particularly in the quantification and analysis of animal behavior. Historically, behavioral quantification relied heavily on tests that were designed with either one or a few readouts in mind. However, the advent of deep learning for computer vision and the development of packages such as DeepLabCut, which enable pose estimation without the need for physical markers, have rapidly expanded the possibilities for non-invasive animal tracking ([Mathis et al., 2020](#)).

By transforming raw video footage into time series data of tracked body parts, these approaches have paved the way for the development of software packages capable of automatically

annotating behavior following a plethora of different approaches, increasing the number of patterns that can be studied per experiment with little burden on the experimenters.

For example, several tools offer options to detect predefined behaviors using supervised machine learning. Along these lines, programs like SimBA (Nilsson et al., 2020), MARS (Segalin et al., 2021), or TREBA (Sun et al., 2021), allow users to label a set of behaviors and train classifiers to detect them in new videos. They employ different labelling schemes which require different amounts of user input, and offer high flexibility in terms of the number of behaviors that can be detected. On the other hand, packages such as B-SOiD (Hsu & Yttri, 2021), VAME (Luxem et al., 2022), and Keypoint-MoSeq (Weinreb et al., 2023), aim for a more exploratory approach that does not require user labelling, but instead relies on unsupervised learning to segment time series into different behaviors. These packages are particularly useful when the user is interested in detecting novel behaviors, or when the number of behaviors is too large to be annotated manually. Moreover, some approaches have been developed to combine the best of both worlds, such as the the A-SOiD active learning framework (Schweihoff et al., 2022), and the semi-supervised DAART (Whiteway et al., 2021). While a thorough discussion on the advantages and disadvantages of each package is beyond the scope of this paper, further information can be found in this recent review (Bordes et al., 2023).

In contrast to other available options, DeepOF offers both supervised and unsupervised annotation pipelines, that allow researchers to test hypotheses regarding experimental conditions such as stress, gene mutations, and sex, in a flexible way (Figure 1).

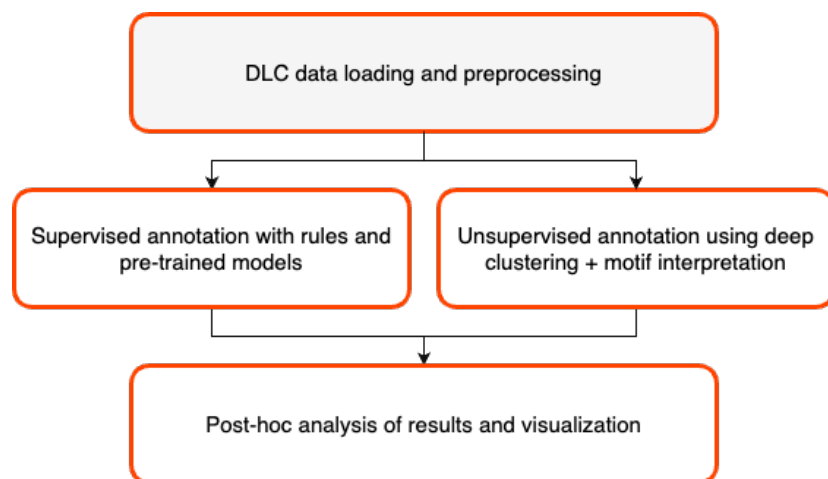


Figure 1: Scheme representing DeepOF workflow. Upon creating a project, DLC data can be loaded and preprocessed before annotating it with either a supervised pipeline (which uses a set of pre-trained models and rule-based annotators) or an unsupervised pipeline, which relies on custom deep clustering algorithms. Patterns retrieved with either pipeline can be passed to downstream post-hoc analysis tools and visualization functions.

The included supervised pipeline uses a series of rule-based annotators and pre-trained machine learning classifiers to detect when each animal is displaying a set of pre-defined behavioral motifs. The unsupervised workflow uses state-of-the-art deep clustering models to extract novel motifs without prior definition. DeepOF then provides an interpretability pipeline to explore what these retrieved clusters are in terms of behavior, which uses both Shapley Additive Explanations (SHAP) (Goodwin et al., 2022) and direct mappings from clusters to video. Moreover, regardless of whether the user chose the supervised annotation pipeline, the unsupervised one, or both, DeepOF provides an extensive set of post-hoc analysis and visualization tools.

When it comes to comparing it to other individual packages that use supervised and unsupervised

annotation, DeepOF stands out in several ways. First of all, it is the first package, to the best of our knowledge, to offer both options. Second, the supervised pipeline in DeepOF follows an opinionated philosophy, in the sense that it provides a set of pre-trained models that cannot be customized, but do not require user labels. This trades flexibility for ease of use, aiming at being a quick exploratory tool that can provide information on key individual and social behaviors with just a few commands. Furthermore, when it comes to the unsupervised pipeline, DeepOF provides three custom deep clustering algorithms capable of segmenting the behavioral time series, as well as the aforementioned built-in interpretability pipeline. If a user runs both pipelines, supervised annotations can be incorporated into this interpretability pipeline in quite a unique way, to detect associations between supervised and unsupervised patterns.

All in all, DeepOF is a comprehensive, end-to-end tool designed to transform DeepLabCut output into relatively quick, exploratory insights on behavioral shifts between experimental conditions, and pinpoint which behaviors are driving them.

Related literature

The DeepOF package has been used to characterize differences in behavior associated with Chronic Social Defeat Stress (CSDS) in mice, as presented in our preprint (currently in revision at the time of writing (Bordes et al., 2022)). There are several other ongoing projects involving the software, although none of them are published to this date.

Acknowledgements

We acknowledge contributions from Felix Agakov and Karsten Borgwardt.

Funding

This project has received funding from the European Union's Framework Programme for Research and Innovation Horizon 2020 (2014-2020) under the Marie Skłodowska-Curie Grant Agreement No. 813533-MSCA-ITN-2018.

References

- Bordes, J., Miranda, L., Müller-Myhsok, B., & Schmidt, M. V. (2023). Advancing social behavioral neuroscience by integrating ethology and comparative psychology methods through machine learning. *Neuroscience & Biobehavioral Reviews*, *151*, 105243. <https://doi.org/10.1016/J.NEUBIOREV.2023.105243>
- Bordes, J., Miranda, L., Reinhardt, M., Brix, L. M., Doeselaar, L. van, Engelhardt, C., Pütz, B., Agakov, F., Müller-Myhsok, B., & Schmidt, M. V. (2022). Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress. *bioRxiv*. <https://doi.org/10.1101/2022.06.23.497350>
- Goodwin, N. L., Nilsson, S. R. O., Choong, J. J., & Golden, S. A. (2022). Toward the explainability, transparency, and universality of machine learning for behavioral classification in neuroscience. *Current Opinion in Neurobiology*, *73*, 102544. <https://doi.org/10.1016/j.conb.2022.102544>
- Hsu, A. I., & Yttri, E. A. (2021). B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. *Nature Communications*, *12*(1). <https://doi.org/10.1038/s41467-021-25420-x>
- Luxem, K., Mocellin, P., Fuhrmann, F., Kürsch, J., Miller, S. R., Palop, J. J., Remy, S., & Bauer, P. (2022). Identifying behavioral structure from deep variational embeddings of

- animal motion. *Communications Biology* 2022 5:1, 5(1), 1–15. <https://doi.org/10.1038/s42003-022-04080-7>
- Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience* 2018 21:9, 21(9), 1281–1289. <https://doi.org/10.1038/s41593-018-0209-y>
- Mathis, A., Schneider, S., Lauer, J., & Mathis, M. W. (2020). A Primer on Motion Capture with Deep Learning: Principles, Pitfalls, and Perspectives. *Neuron*, 108(1), 44–65. <https://doi.org/10.1016/j.neuron.2020.09.017>
- Nilsson, S. R., Goodwin, N. L., Choong, J. J., Hwang, S., Wright, H. R., Norville, Z. C., Tong, X., Lin, D., Bentzley, B. S., Eshel, N., McLaughlin, R. J., & Golden, S. A. (2020). Simple Behavioral Analysis (SimBA) – an open source toolkit for computer classification of complex social behaviors in experimental animals. *bioRxiv*. <https://doi.org/10.1101/2020.04.19.049452>
- Schweihoff, J. F., Hsu, A. I., Schwarz, M. K., & Yttri, E. A. (2022). A-SOiD, an active learning platform for expert-guided, data efficient discovery of behavior. *bioRxiv*. <https://doi.org/10.1101/2022.11.04.515138>
- Segalin, C., Williams, J., Karigo, T., Hui, M., Zelikowsky, M., Sun, J. J., Perona, P., Anderson, D. J., & Kennedy, A. (2021). The Mouse Action Recognition System (MARS) software pipeline for automated analysis of social behaviors in mice. *eLife*, 10, e63720. <https://doi.org/10.7554/eLife.63720>
- Sun, J. J., Kennedy, A., Zhan, E., Anderson, D. J., Yue, Y., & Perona, P. (2021, June). Task Programming: Learning Data Efficient Behavior Representations. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. <https://doi.org/10.1109/cvpr46437.2021.00290>
- Weinreb, C., Osman, M. A. M., Zhang, L., Lin, S., Pearl, J., Annapragada, S., Conlin, E., Gillis, W. F., Jay, M., Ye, S., Mathis, A., Mathis, M. W., Pereira, T., Linderman, S. W., & Datta, S. R. (2023). Keypoint-MoSeq: parsing behavior by linking point tracking to pose dynamics. *bioRxiv*. <https://doi.org/10.1101/2023.03.16.532307>
- Whiteway, M. R., Schaffer, E. S., Wu, A., Buchanan, E. K., Onder, O. F., Mishra, N., & Paninski, L. (2021). Semi-supervised sequence modeling for improved behavioral segmentation. *bioRxiv*. <https://doi.org/10.1101/2021.06.16.448685>

5 Characterizing CSDS using automatically annotated motion tracking data

5.1 Overview

Once the presented algorithms were implemented and packaged in DeepOF, there came the time to apply them to a real-world dataset. The model of choice, as introduced in chapter 1, was Chronic Social Defeat Stress (CSDS), a widely adopted animal model on chronic stress and depression research. This acted as a positive control, where we had a clear idea of the expected shifts to detect, which would serve as validation for the provided package. Moreover, as significant symptoms of MDD are the deterioration of social functionality and a decline in social motivation, it also served as a good platform to test the detection of social interaction alterations in several ways.

Along these lines, the current chapter includes our paper titled *Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress*, published in *Nature Communications* [75]. In short, we here show how DeepOF can identify distinct stress-induced social and individual behavioral patterns, relying on both the annotation of pre-defined traits, and on unsupervised segmentation using the models presented in chapter 3 ¹. Moreover, analyzing how global embedding shifts evolve over time reveals how these patterns are particularly noticeable at the onset of a novel social interaction, though they tend to diminish over time due to habituation in social settings.

We apply DeepOF to a variety of experimental settings, including single-animal open fields, social interaction (for which we use both single and multi-animal embeddings), and social avoidance tasks, and compare and interpret how the detected patterns vary across settings. Furthermore, while traditional social avoidance tasks (a set of univariate measures introduced in chapter 1) can detect stress-induced social behavioral differences, both supervised and unsupervised DeepOF pipelines offer a more comprehensive and detailed profile, which in addition requires lower experimental effort. Last but not least, a comprehensive statistical and visual interpretation of retrieved clusters is included.

5.2 Contribution to the work

I was the sole contributor of the DeepOF package, whose implementation was presented in the previous chapter and application is presented here. Moreover, I had the leading role in analyzing all the data presented in this chapter. Main figures 1 (except for mice drawings), 6, 7, as well as supplemental figures 1, 2, and 5–17 were conceived, designed, and coded by me. I did not take part in any of the animal experiments involved. I also wrote the full text in the manuscript (together with co-lead Joeri Bordes). First author order in the authors' list was randomized.


¹As introduced already in previous chapters, DeepOF offers three different encoder-decoder architectures (recurrent, TCN, transformers) and three time series segmentation approaches (VaDE, VQVAE, Contrastive+HMM). All results presented in this paper, however, relied on VaDE models with graph-like inputs and recurrent encoders/decoders. Results using other models did not substantially modify what was obtained, except for graph-input VQVAEs, which often diverged during training.

Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress

Received: 19 September 2022

Accepted: 7 July 2023

Published online: 18 July 2023

 Check for updates

Joeri Bordes^{1,6}, Lucas Miranda^{2,3,6}, Maya Reinhardt¹, Sowmya Narayan^{1,3}, Jakob Hartmann⁴, Emily L. Newman⁴, Lea Maria Brix^{1,3}, Lotte van Doeselaar^{1,3}, Clara Engelhardt¹, Larissa Dillmann¹, Shiladitya Mitra¹, Kerry J. Ressler⁴, Benno Pütz², Felix Agakov⁵, Bertram Müller-Myhsok²✉ & Mathias V. Schmidt¹✉

Severe stress exposure increases the risk of stress-related disorders such as major depressive disorder (MDD). An essential characteristic of MDD is the impairment of social functioning and lack of social motivation. Chronic social defeat stress is an established animal model for MDD research, which induces a cascade of physiological and behavioral changes. Current markerless pose estimation tools allow for more complex and naturalistic behavioral tests. Here, we introduce the open-source tool DeepOF to investigate the individual and social behavioral profile in mice by providing supervised and unsupervised pipelines using DeepLabCut-annotated pose estimation data. Applying this tool to chronic social defeat in male mice, the DeepOF supervised and unsupervised pipelines detect a distinct stress-induced social behavioral pattern, which was particularly observed at the beginning of a novel social encounter and fades with time due to habituation. In addition, while the classical social avoidance task does identify the stress-induced social behavioral differences, both DeepOF behavioral pipelines provide a clearer and more detailed profile. Moreover, DeepOF aims to facilitate reproducibility and unification of behavioral classification by providing an open-source tool, which can advance the study of rodent individual and social behavior, thereby enabling biological insights and, for example, subsequent drug development for psychiatric disorders.

Stress is an essential aspect of our daily lives, which contributes to our mood and motivation. However, exposure to severe stress can have negative consequences and has become an increasing burden on society. In particular, stress-related disorders, such as major depressive disorder (MDD), have been steadily on the rise for the last decade¹. Our understanding of the behavioral and neurobiological mechanisms

related to MDD is limited, which is part of the reason for the only moderate success of current drug treatments². MDD is a complex and heterogeneous disorder, and its classification is dependent on a widespread set of symptoms. An important characteristic of MDD is the impairment of social functioning and lack of social motivation, which can lead to social withdrawal from society in extreme cases³. In

¹Research Group Neurobiology of Stress Resilience, Max Planck Institute of Psychiatry, 80804 Munich, Germany. ²Research Group Statistical Genetics, Max Planck Institute of Psychiatry, 80804 Munich, Germany. ³International Max Planck Research School for Translational Psychiatry (IMPRS-TP), 80804 Munich, Germany. ⁴Department of Psychiatry, Harvard Medical School, McLean Hospital, Belmont, MA 02478, USA. ⁵Pharmatics Limited, Edinburgh EH16 4UX Scotland, UK. ⁶These authors contributed equally: Joeri Bordes, Lucas Miranda. ✉e-mail: bmm@psych.mpg.de; mschmidt@psych.mpg.de

addition, disturbances in social behavior are an important risk factor for developing MDD, as poor social networks are linked to lowered mental and physical health^{4,5}. The impact of social interactions was highlighted during the COVID-19 pandemic, where a substantial part of society experienced little to no social interactions for a sustained period. An increasing number of studies are now reporting the enormous impact of the pandemic, emphasizing a dramatic increase in the prevalence of stress-related disorders, in particular MDD^{6,7}. Unfortunately, there is still a lack of awareness of the importance of social interactions and their role in stress-related disorders. Therefore, it is crucial to increase the understanding of the biological and psychological mechanisms behind MDD, and the influence of social behavior on the development of MDD.

Along these lines, animal models have an important role in MDD research. Although unable to recreate the exact nature of the disorder in humans, they provide a controlled environment where symptoms of MDD can be investigated^{8,9}. The well-established chronic social defeat stress (CSDS) paradigm is continuously used for studying symptoms of MDD in animals^{10,11}. In the CSDS model, mice are subjected daily to severe physical and non-physical stressors from aggressive mice for several weeks, which results in the chronic activation of the physiological stress response system, leading to bodyweight differences, enlarged adrenals, and elevated levels of corticosterone¹². In addition, animals subjected to CSDS show stress-related behaviors such as social avoidance, anhedonia, reduced goal-directed motivation, and anxiety-like behavior^{10,13–16}. Especially CSDS-induced social avoidance behavior, which is the avoidance of a novel conspecific, is a recognized phenomenon that is used to investigate the social neurobiological mechanisms related to chronic stress exposure and stress-related disorders^{11,17,18}.

Currently, several social behavioral tasks can assess different constructs of social behavior, particularly the social avoidance task¹⁸. It is important that these behavioral tasks are conducted with control over the environment to investigate the effects of external stimuli, such as stress exposure. For decades there has been a trend to standardize and simplify these tests to allow for greater comparability and higher throughput. Unfortunately, this has led to an oversimplification of the social behavioral repertoire and increased the risk for cross-over effects by other types of behavior, such as anxiety-related behavior. Moreover, due to limitations in tracking software, the analysis of the interaction between multiple freely moving animals remained difficult, which further limited the complexity of the behavioral assessment. Social behavior is a complex behavioral construct, which relies on many different types of behavioral interactions, that often are too complicated, time-intensive, and repetitive to assess manually^{19–21}. Ultimately, this can lead to poor reproducibility of the social behavioral construct, as observed for social approach behavior²².

The current advancement in automatically annotated behavioral assessment, however, allows for high-throughput analysis using pose estimation, involving both supervised classification (intending to extract pre-defined and characterized traits) and unsupervised clustering (which aims to explore the data and extract patterns without external information)^{23–28}. Importantly, the open-source tool DeepLabCut has provided a robust and easily accessible system for deep-learning-based motion tracking and markerless pose estimation^{29,30}. The use of supervised classification, by defining the behavioral patterns of interest a priori, is a powerful tool that simplifies the analysis by using predefined relevant behavioral constructs without losing the complexity of social behavior. Furthermore, recent studies have shown the value of unsupervised clustering in addition to a supervised analysis, which can reveal novel and more complex structures of behavior^{19,26,31–33}. By acting in a more exploratory fashion, these practices can not only assist the discovery of novel traits but also direct researchers toward the main behavioral axes of variation across cohorts of interest. In addition, both the supervised and unsupervised

analysis approaches can provide more transparency for the behavioral definition and can easily be shared via online repositories, which contributes to a more streamlined definition of behavior across different labs^{21,34}. These computational tools can elevate the current understanding of the influences of stress exposure on behavior, by increasing the resolution of the observed behavioral output³⁵.

Therefore, the current study provides an application of our open-source tool DeepOF³⁶, which enables users to delve into the individual and social behavioral profiles of mice using DeepLabCut-annotated pose estimation data (Fig. 1). DeepOF provides two main workflows; a supervised behavioral analysis pipeline, which applies a set of annotators and pre-trained classifiers to detect defined individual and social traits, and an unsupervised analysis pipeline, capable of embedding the motion-tracking data of one or more animals in a latent behavioral space, pointing toward differences across experimental conditions without any label priming. Furthermore, DeepOF can retrieve unsupervised clusters of behavior that can be compared across conditions and therefore hint at previously unrecognized behavioral patterns that trigger new hypotheses. We describe a distinct social behavioral profile following CSDS in mice that can be recapitulated with both supervised and unsupervised workflows. Moreover, the current study observes a clear state of arousal upon exposure to a novel social conspecific that fades over time, which provides crucial insights for the quantification of optimal behavioral differences across time and experimental conditions.

Results

The supervised pipeline provided by DeepOF yields generalizable annotations

As expected, all rule-based behaviors show high performance when compared to manual labeling, which constitutes an argument in favor of simple behavioral tagging (Supplementary Fig. 1).

When evaluating the performance of the huddle classifier, balanced accuracy in the training set (0.78 ± 0.005) was marginally higher than in both validation settings (suggesting no overfitting), and performance on the internal validation (0.75 ± 0.046) was not significantly higher than performance on the external validation (0.75 ± 0.04) suggesting excellent generalization to new datasets (independent samples t-test: $T(7.34) = -0.03$, $p = 0.51$, Supplementary Fig. 2A). In addition, pseudo-labeling conducted on the external dataset showed a strong and significant correlation between total behavior duration across manual and predicted labels (Supplementary Fig. 2B). Finally, the SHAP analysis of the deployed classifier revealed low head movement, low spine stretch, low body area, and low locomotion speed as the most important features of the model, which goes in line with the accepted definition of the behavior (Supplementary Fig. 2C).

The physiological and behavioral hallmarks of stress are reproduced by CSDS

The CSDS paradigm was performed to maintain stress exposure for several weeks (Fig. 2A), which induced dysregulation of the hypothalamic-pituitary-adrenal axis (HPA-axis) and a stress-related behavioral profile. Male mice that were subjected to CSDS showed clear hallmarks of stress exposure, as observed by a significant increase in body weight during the stress paradigm, which was especially apparent towards the end of the stress (Fig. 2B, C), an increase in relative adrenal weight (Fig. 2D), reduced locomotion and time spent in the inner zone of the OF (Fig. 2E, F), and a significantly reduced SA-ratio in the SA task (Fig. 2G). Notably, no bodyweight difference was observed at the beginning of the CSDS paradigm (Fig. 2B).

Further exploration of the OF data using PCA across four 2.5 min consecutive time bins showed that all time bins were significantly different from each other, suggesting that they all should be included in further behavioral analysis of the OF data (Supplementary Fig. 3A,

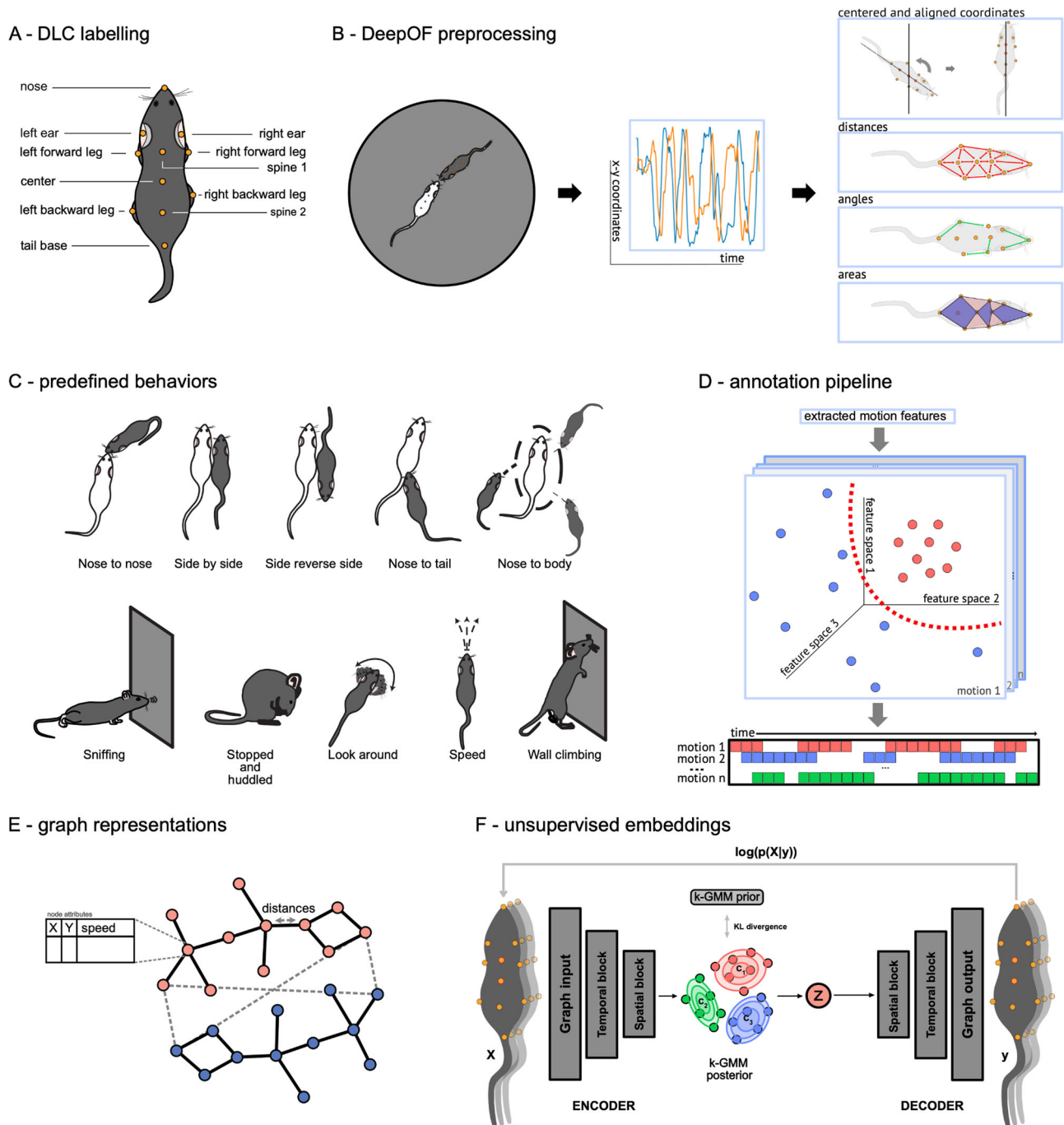


Fig. 1 | DeepOF workflow. **A** 11 labels were tagged on each annotated mouse using DeepLabCut. **B** DeepOF preprocessing pipeline. One or two mice (a C57Bl/6N experimental subject and a CD1 social companion depending on the dataset) were tagged using the provided DeepLabCut models. After tracking body parts with DeepLabCut, DeepOF was used to smooth the retrieved trajectories, interpolate outliers, and extract features (including coordinates, distances, angles, areas, speeds and accelerations). **C** Set of predefined behaviors that the DeepOF supervised pipeline can retrieve. These include dyadic motifs (such as nose-to-nose contacts) and individual motifs (such as climbing), which are reported individually for all tracked mice. The stopped-and-huddled classifier²⁸ is abbreviated as “huddle” in DeepOF output (not to be confused with group huddling behavior⁶⁷). **D** Schematic representation of the supervised pipeline in DeepOF. A set of extracted motion features (only three dimensions are shown for visualization purposes) are fed to a set of rule-based annotators and pre-trained classifiers, which report the presence of each behavioral trait at each time by learning how the corresponding trait is distributed in the feature space (red dots). The set of

classifiers then yields a table indicating the presence of each motif across time, which can be used for further analysis. Note that annotators are not necessarily mutually exclusive, as several predictors can be triggered at the same time. **E** Graph representation of animal trajectories used by DeepOF in the unsupervised pipeline. All 11 body parts per animal are connected using a pre-designed (but customizable) adjacency matrix. Nodes are annotated with x, y coordinates and speed of each body part at each given time, and edges with the corresponding distances. This representation can also handle multi-animal settings, where the graphs of individual animals are connected with nose-to-nose, nose-to-tail, and tail-to-tail edges. **F** Schematic representation of the deep neural network architecture used for the unsupervised clustering of behavior. Data is embedded with a sequence-aware spatio-temporal graph encoder, and clustered at the same time by selecting the argmax of the likelihood of the components of a mixture-of-Gaussians latent posterior. Unidirectional black arrows indicate forward propagation, and gray arrows indicate the reconstruction and KL divergence terms of the loss function, the latter of which minimizes the distance to an also mixture-of-Gaussians prior.

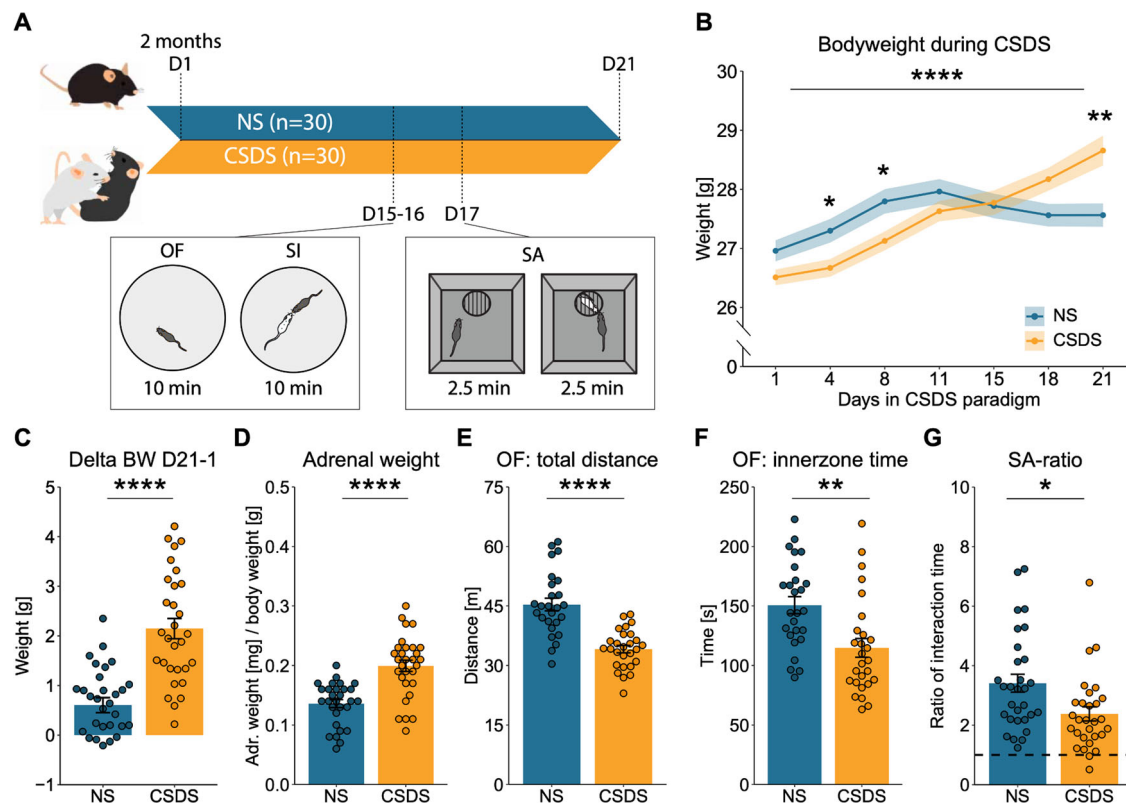


Fig. 2 | Classical hallmarks for chronic social defeat stress. **A** Experimental timeline for the CSDS paradigm and behavioral testing, including the open field (OF) and social interaction (SI) task on day 15–16 (animals were divided between the two days) and social avoidance (SA) task on day 17. **B** Significant increase of body weight after CSDS exposure (two-way ANOVA: within-subject effect of time: $F(6,406) = 13.58, p = 4.59 \times 10^{-14}$, as well as time \times condition interaction effect: $F(6,406) = 6.13, p = 3.65 \times 10^{-6}$, but no between-subject effect on condition: $F(1,406) = 0.20, p = 0.653$). Post-hoc analysis with Benjamini Hochberg revealed no significant difference on day 1, 11, 15, and 18, but there was a significant difference on day 4 ($T(1,58) = 6.36, p = 0.033$), 8 ($T(1,58) = 6.55, p = 0.033$), and 21 ($T(1,58) = 11.57, p = 0.007$). **C** The delta body weight during the CSDS paradigm (day 21–day 1) was

significantly increased in CSDS-exposed animals (Two-tailed independent samples t-test: $T(58) = -6.09, p = 9.8 \times 10^{-8}$). **D** Increase of relative adrenal weight after CSDS exposure (Two-tailed independent samples t-test: $T(57) = -5.44, p = 1.15 \times 10^{-6}$). **E** The total locomotion in the OF was reduced after CSDS exposure (Two-tailed independent samples t-test: $T(51) = 6.15, p = 1.18 \times 10^{-7}$). **F** The inner zone time in the OF was reduced after CSDS exposure (Two-tailed independent samples t-test: $T(51) = 3.37, p = 0.0015$). **G** The SA-ratio was reduced in the SA task after CSDS exposure (Two-tailed wilcoxon test: $W = 617, p = 0.006$). The timeline and bar graphs are presented as mean \pm standard error of the mean and all individual samples as points. $N = 30$ for NS and CSDS for (B–G). Source data are provided as a Source Data file.

B). The OF PCA between conditions revealed a significant difference and showed the importance of the OF parameters, in which total distance, look-around, and sniffing came out as the top contributing behaviors (Supplementary Fig. 3C, D). A significant stress effect was observed for the total distance, look-around, and inner-zone time throughout the different time bins, whereas sniffing was altered, but not in all time bins (Supplementary Fig. 3E–J). Importantly, even though a stress-induced effect can be found in the OF task, a general habituation effect to the OF in both NS and CSDS can be observed, as total distance reduces over time, while look-around and sniffing increase. The successful habituation to the novel environment is crucial for the subsequent SI task to allow full attention to the novel social conspecific (Supplementary Fig. 3E–G).

DeepOF social behavioral classifiers show a stronger PCA separation for stress exposure than social avoidance

The social behavioral pattern during the SI task was investigated in four non-overlapping time bins of 2.5 min each to match the time frame in the SA task. Principal component analysis (PCA) was performed to show the difference between time bins in the social behavioral profile regardless of the animal's stress condition (Fig. 3A). Interestingly, the PCA showed a significant effect between the time bins, in which the first 2.5 min time bin was significantly different from the subsequent ones (5, 7.5, and 10 min). In contrast, the subsequent time bins did not show variation between one another (Fig. 3B). This

suggests that the different time bins in the SI task are an important variable, and that the first 2.5 min time bin should be specifically investigated. Next, the SA and SI tasks were compared on their ability to distinguish between NS and CSDS animals. PCAs were performed for the SA task (Fig. 3C) and the 2.5 min time bin SI data (Fig. 3D, E), both of which showed a significant difference between the conditions in the principal component (PC) 1 eigenvalues (Fig. 3C–E). However, the SI task showed a clearer separation of the conditions than the SA task, suggesting that the SI task is a more powerful tool for identifying stressed animals than the SA task. In addition, the PC1 top contributing behaviors for the 2.5 min time bin SI data were calculated using the corresponding rotated loading scores (Fig. 3F). The top five contributing behaviors were reported as essential behaviors for identifying the stressed phenotype, which consisted of B-huddle, B-look-around, B-nose-to-tail, B-speed, and B-nose-to-body from the C57Bl/6N animal, whereas the other behaviors within the top 10 were either contributing to the CD1 animal or had a low rotated loading score (Fig. 3F). Here, “B-” indicates behaviors related to or initiated by the C57Bl/6N animals, whereas “W-” refers to the CD1.

DeepOF social behavioral classifiers are strongly altered by CSDS

Next, the influence of the CSDS on the top five contributing behaviors in the SI task was investigated. In accordance with the PCA time bin analysis, a clear stress-induced effect was observed, with elevated

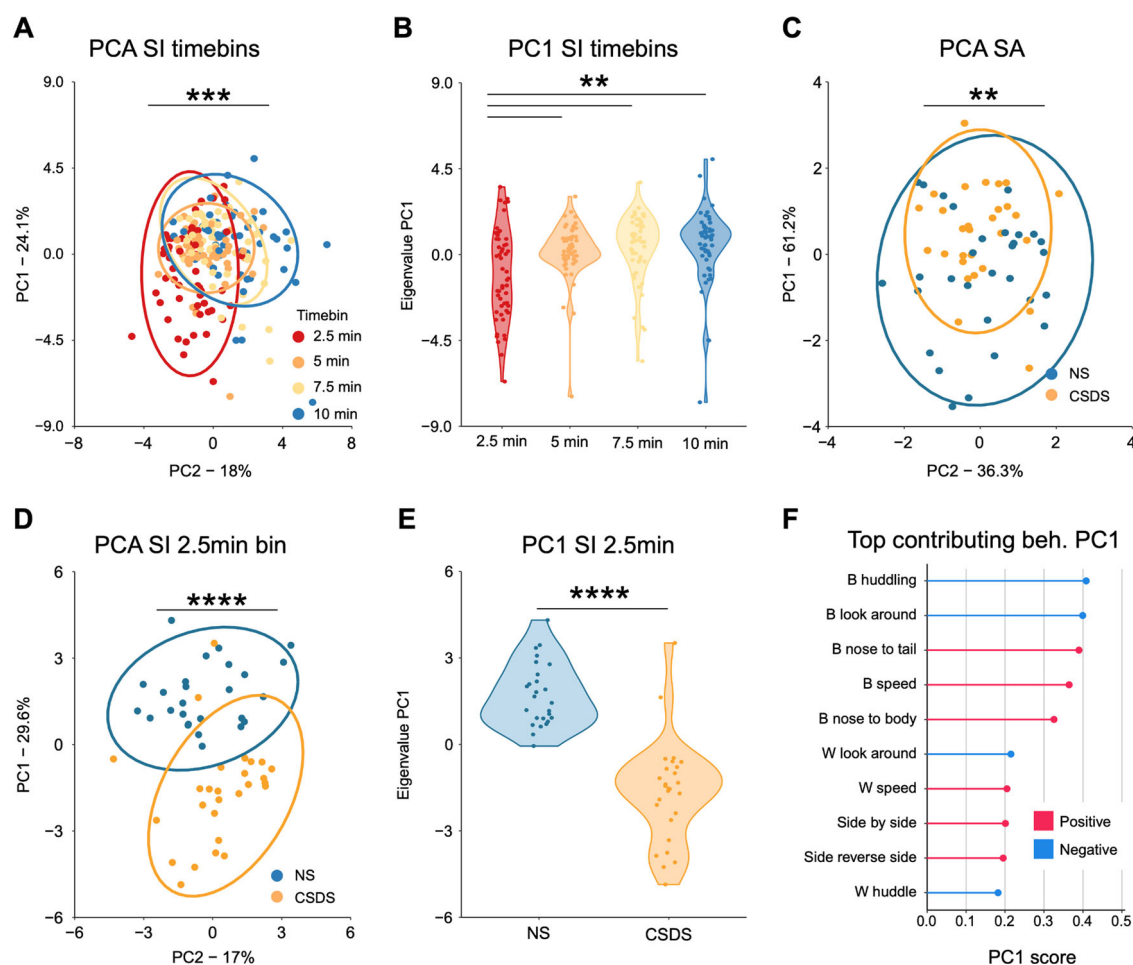


Fig. 3 | Social interaction binning yields more separable PCA projections than the social avoidance task. **A** In the SI data a PCA revealed that the first 2.5 min time bin is significantly different from the other time bins. (Kruskal-Wallis test: $H(3) = 19.90$, $p = 0.0002$. **B** The PC1 eigenvalues of the SI time bin PCA. Post-hoc Wilcoxon: 2.5 min vs. 5 min ($W = 957$, $p = 0.01$), 2.5 min vs. 7.5 min ($W = 860$, $p = 0.0018$), 2.5 min vs. 10 min ($W = 811$, $p = 0.0011$). **C** The SA task PCA showed a significant difference in the PC1 eigenvalues between conditions. The PCA data consisted of the SA-ratio, total time spent with the non-social stimulus, and total time spent with the social stimulus. Two-tailed independent samples t -test: $T(57) = -2.84$, $p = 0.006$. **D** The SI 2.5 min time bin PCA showed a significant difference in the PC1 eigenvalues between conditions. The PCA data consisted of all the SI DeepOF behavioral classifiers, as listed in Fig. 1C. Two-tailed independent

samples t -test: $T(51) = 8.28$, $p = 5.39e-11$. **E** The PC1 eigenvalues of the 2.5 min time bin SI task. **F** The top contributing behaviors of the SI 2.5 min time bin in PC1 using the corresponding rotated loading scores. The top five behaviors were reported as the essential behaviors for identifying stress exposure (B-huddle (-0.41), B-look-around (-0.40), B-nose-to-tail (0.39), B-speed (0.36), B-nose-to-body (0.33). “B-” indicates C57Bl/6N behaviors and “W-” indicates CD1 behaviors. The PCA graphs (Fig. 3A, C, D) are provided with a 95% confidence ellipse and all individual samples as points. Further PC1 analyses (Fig. B, E) are represented with a violin plot and all individual samples as points. In Fig. 3F the absolute score of the PC1 value is represented by the point. $N = 26$ for NS and $n = 27$ for CSDS in (A, B, D–F) and $n = 30$ for NS and CSDS in (C). Source data are provided as a Source Data file.

duration in the CSDS animals for B-look-around (Fig. 4A, B) and B-huddle (Fig. 4C, D), while lowered for the B-speed (Fig. 4E, F), B-nose-to-tail (Fig. 4G, H), and B-nose-to-body (Fig. 4I, J). The total duration per time bin for the top contributing behaviors showed the strongest CSDS-induced effect in the 2.5 min time bin data (supplemental Fig. 4, timeline graphs), compared to the 5, 7.5, and 10 min time bins. In addition, supplemental Fig. 4 shows the 10 min total duration and time bin analyses for all other DeepOF behavioral classifiers, in which a significant stress effect is observed for B-sniffing, B-wall-climbing, and Side-by-side.

Z-score for DeepOF social interaction correlates with Z-score for stress physiology

The Z-score of stress physiology was calculated using the relative adrenal weight and body weight on day 21 of the CSDS. The stress physiology Z-score provides a strong CSDS profiling tool and was used for correlation analysis between the SA and SI tasks. Even though the behavioral and physiological readouts were not obtained at the same

time, the former can be used as a proxy of the impact of the stress exposure, and are expected to be stable during the last week of the CSDS pipeline. No significant correlation was observed between the Z-score of stress physiology and the SA ratio (Fig. 5A). Subsequently, the Z-score of SI was calculated by using the 2.5 min time bin of the top five contributing behaviors in the SI task (Fig. 4). Stress physiology and SI Z-score showed a significant positive correlation (Fig. 5B), which indicates that the SI Z-score provides a stronger tool for CSDS profiling compared to the SA ratio. Next, correlation analyses were performed between the Z-score of SI and all other behavioral and physiological measurements which indicated a strong correlation with several OF parameters. Highly affected OF parameters, such as speed, distance, inner zone entries, and look-around might be directly related to social anxiety and warrant further investigation. Interestingly, no correlation with the SA ratio was observed (Fig. 5C).

Notably, the SA task is extensively used to distinguish resilient and susceptible animals in the CSDS paradigm^{10,17}, and depending on the protocol and stress severity this can give a distinction between

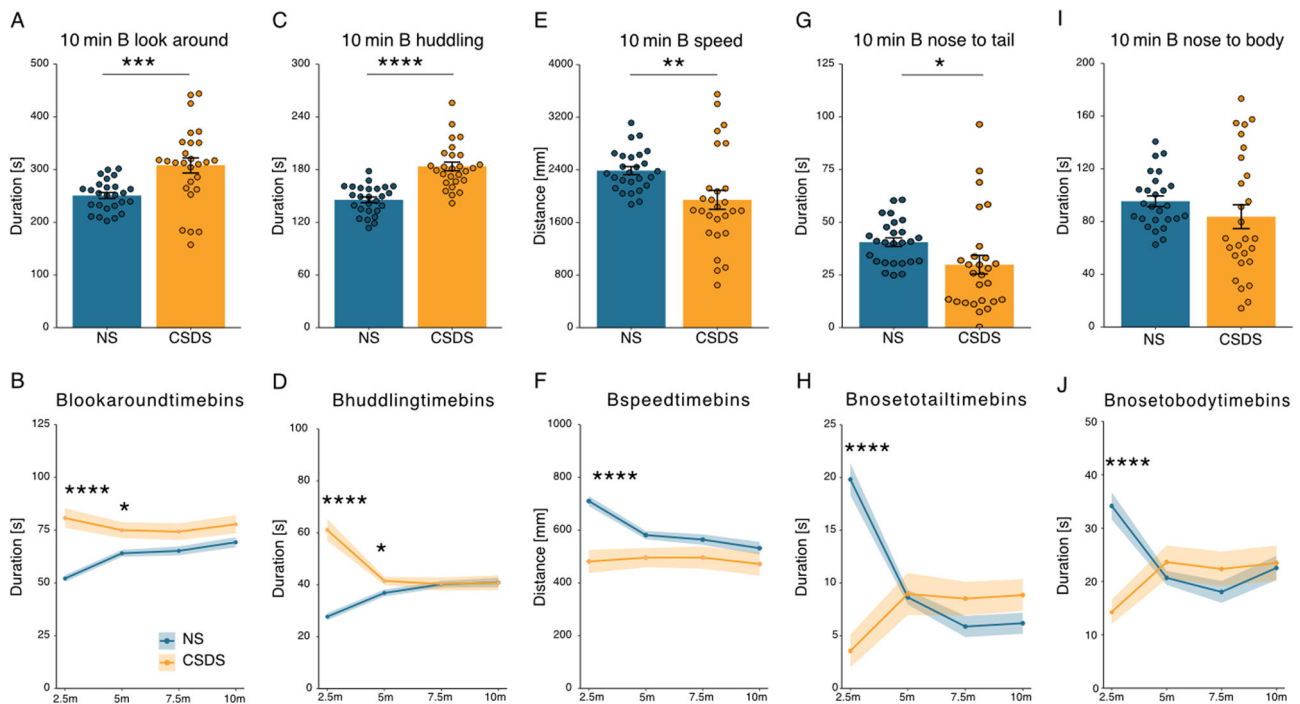


Fig. 4 | Top contributing behaviors in the social interaction task for 10 min total duration and time bins. **A** The total duration of B-look-around. Two-tailed Welch: $T(34.1) = -3.71$, $p = 0.0007$. **B** Time bin for B-look-around. Benjamini Hochberg (BH) posthoc for the 2.5 min time bin: ($T(51) = 33.46$, $p = 1.78e-6$) and the 5 min time bin ($T(51) = 6.84$, $p = 0.024$), but not for the 7.5 and 10 min time bins ($p = 0.067$, $p = 0.093$, respectively), two-way ANOVA: condition effect: $F(1,208) = 37.45$, $p = 4.59e-9$, time effect: $F(1,208) = 4.02$, $p = 0.046$, and condition \times time effect: $F(1,208) = 8.87$, $p = 0.003$. **C** The total duration of B-huddle. Two-tailed independent samples t -test: $T(51) = -6.40$, $p = 4.8e-8$. **D** Time bin for B-huddle. Wilcoxon posthoc for the 2.5 min time bin ($W(26,27) = 63.5$, $p = 1.3e-6$), and the 5 min time bin ($W(26,27) = 204$, $p = 0.018$), but not for the 7.5- and 10 min time bins ($p = 0.52$, $p = 0.52$, respectively), Kruskal-Wallis: 2.5 min: $p = 1.25e-6$, 5 min: $p = 0.018$, 7.5 min: $p = 0.51$, and 10 min: $p = 0.51$. **E** The total duration of B-speed. Two-tailed Welch: $T(35.04) = 2.84$, $p = 0.0074$. **F** Time bin for B-speed. BH posthoc for the 2.5 min time bin ($T(51) = 22.41$, $p = 7.16e-5$), but not for the 5-, 7.5-, and 10 min time bins

($p = 0.076$, $p = 0.20$, $p = 0.24$, respectively), two-way ANOVA: condition effect: $F(1,208) = 22.60$, $p = 3.72e-6$, time effect: $F(1,208) = 7.51$, $p = 0.007$, and condition \times time effect: $F(1,208) = 6.34$, $p = 0.013$. **G** The total duration of B-nose-to-tail. Two-tailed Welch: $T(36.70) = 2.18$, $p = 0.036$. **H** Time bin for B-nose-to-tail. Wilcoxon posthoc for the 2.5 min time bin ($W(26,27) = 660$, $p = 1.5e-7$), but not for the 5-, 7.5-, and 10 min time bins ($p = 0.19$, $p = 0.49$, $p = 0.49$, respectively), Kruskal-Wallis: 2.5 min: $p = 1.43e-7$, 5 min: $p = 0.18$, 7.5 min: $p = 0.48$, 10 min: $p = 0.48$. **I** The total duration of B-nose-to-body. Welch: $T(35.85) = 1.18$, $p = 0.24$. **J** Time bin for B-nose-to-body. Wilcoxon posthoc for the 2.5 min time bin ($W(26,27) = 626.5$, $p = 3.97e-6$), but not for the 5, 7.5 and 10 min time bins ($p = 0.85$, $p = 0.85$, $p = 0.85$, respectively), Kruskal-Wallis: 2.5 min: $p = 3.8e-6$, 5 min: $p = 0.85$, 7.5 min: $p = 0.85$, 10 min: $p = 0.85$. The timeline and bar graphs are presented as mean \pm standard error of the mean and all individual samples as points. $N = 26$ for NS and $n = 27$ for CSDS in (A–J). Source data are provided as a Source Data file.

resilient and susceptible animals (Fig. 5D–F). Interestingly, while clearly differentiating affected and non-affected individuals, the DeepOF module does not find a distinction between SA-ratio-defined susceptibility and resiliency on the 2.5 min bin SI DeepOF behavioral classifiers (Fig. 5G–M), indicating that the DeepOF behavioral classifiers represent a unique and distinguished set of resilience-linked phenotypes.

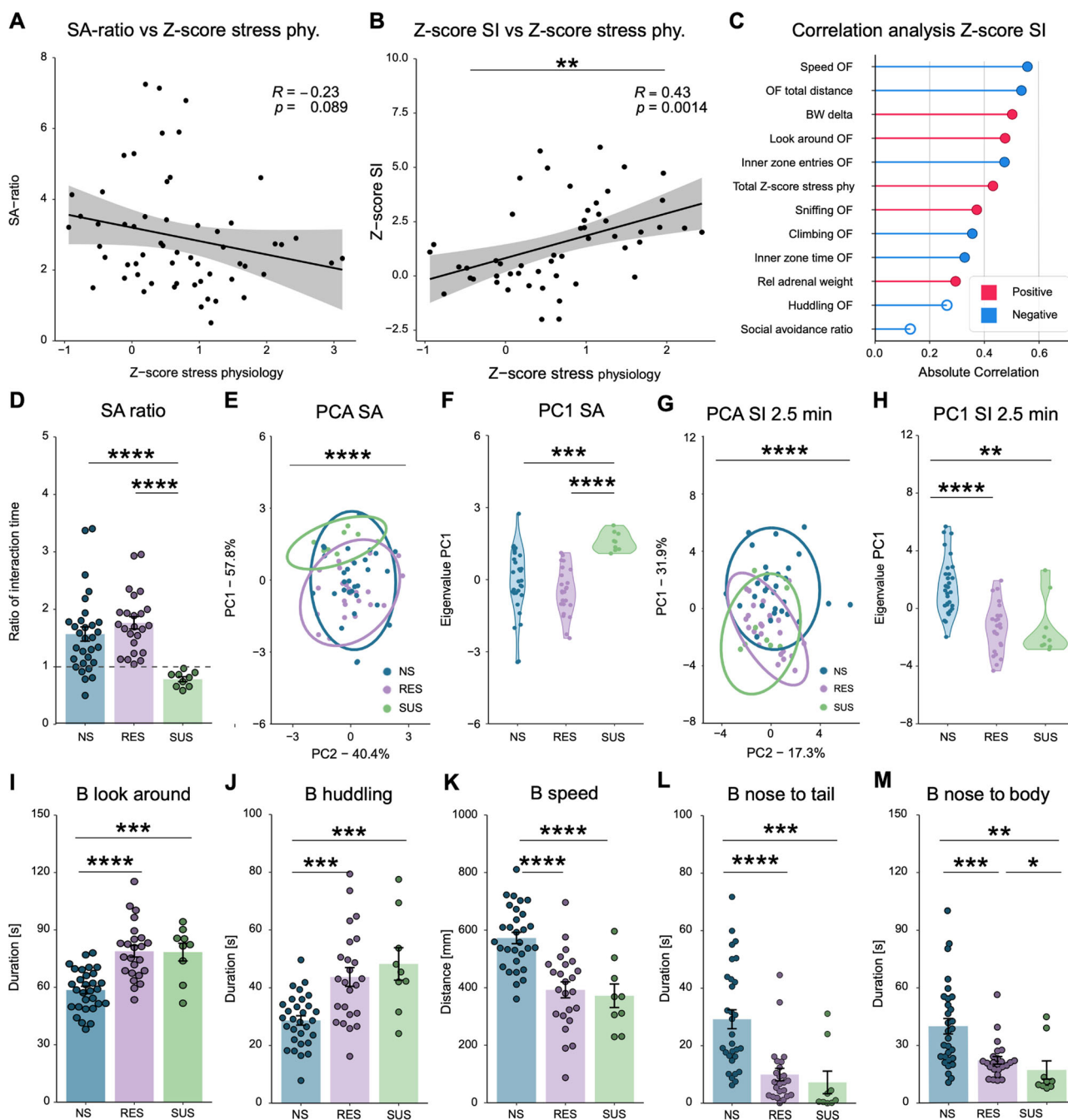
The DeepOF unsupervised pipeline can be flexibly applied across different experimental settings

The unsupervised pipeline within DeepOF was applied to three datasets and four settings. These included both single and multi-animal embeddings on the SI dataset, single-animal embeddings on the OF dataset, and single-animal embeddings on the SA dataset. When applying this workflow to a new dataset, the number of clusters is a hyperparameter the user must tune. In this study, an optimal solution was found by selecting the number of clusters that explains the largest difference between experimental conditions (in terms of the area under the ROC curve of a classifier to distinguish between them, see methods for details). While DeepOF could be used to describe the behavioral space of a single condition, this model selection procedure aims at maximizing the power to detect behavioral differences between experimental conditions. An optimum of 10 clusters was measured for both single- and multi-animal SI settings (Fig. 6A and

Supplementary Fig. 5A), whereas the single-animal OF setting showed an optimum of 11 clusters (Supplementary Fig. 6A), and the SA setting of 17 clusters (Supplementary Fig. 7A). Timepoint UMAP projections of the latent space depicting all clusters can be found in Fig. 6B, and Supplementary Figs. 5B, 6B, and 7B for all four settings, respectively.

DeepOF can quantify behavioral differences over time in an unsupervised way

Once the number of clusters was fixed, the stress-induced phenotype was investigated over time in both SI and OF settings. SA was excluded of this analysis due to the shorter length of the videos (2.5 min), in which no decay of arousal should be observed in the animals. To this end, a growing time window spanning an increasing number of sequential seconds was analyzed. For each analysis, the discriminability between conditions was tested by evaluating the performance of a linear classifier to distinguish between them in the global animal embedding space, for which each experiment is represented by a vector containing the time spent per cluster (see methods for details). The bin size for which discriminability was maximized was then selected as optimal and used for further analysis. In this case, we observed an optimum of 126 and 124 s for the single-animal and multi-animal SI tasks respectively, indicating that differences between conditions are maximized early in the 10-min-long experiments, which is compatible with habituation. Furthermore, performance across



consecutive, non-overlapping bins retaining the optimal size was also reported (Fig. 6C and Supplementary Fig. 5C). Here, decaying performance across bins in the SI setting is also compatible with a state of arousal, where conditions become less distinguishable over time after the behavior of the C57Bl/6N mice becomes less influenced by novelty. The largest difference between NS and CSDS animals can thus be observed during this period. In line with this finding, the optimal distance in the single animal OF data was reached at 595 s, suggesting that no binning is necessary since behavior between conditions remains consistently distinguishable across the videos (Supplementary Fig. 6C).

Interestingly, global animal embeddings show a clearer separation between conditions in both single and multi-animal embeddings for the SI setting (Fig. 6D and Supplementary Fig. 5D), whereas the difference is milder in the OF setting, as the projected distributions are less separable (Supplementary Fig. 6D). In the SA setting, projections

show, as expected, a higher separation between conditions in trial two, which includes the engaged conspecific (Supplementary Fig. 7C, D).

These global embeddings also capture how distributions merge over time in the SI settings, as the behavioral profiles of NS and CSDS mice become closer (Supplementary Figs. 8, 9).

Individual unsupervised clusters reveal differences in behavior enrichment

Going beyond global differences in behavior, the aggregated embeddings depicted so far are the result of summarizing the expression of the set of detected behavioral clusters. Once obtained, DeepOF enables the user to test the differential expression between conditions. To this end, the time spent on each cluster across all videos for each condition is recorded for each time bin. Importantly, DeepOF has no knowledge of the assigned animal conditions at the time of training and assigning clusters.

Fig. 5 | Z-score correlation analysis and the exploration of susceptibility and resiliency. **A** Pearson correlation analysis between the SA-ratio and the Z-score of stress physiology ($R = -0.23$, $p = 0.089$). **B** Pearson correlation analysis between the SI task 2.5 min time bin top five contributing behaviors and the Z-score of stress physiology ($R = 0.43$, $p = 0.0014$). **C** Pearson correlation analyses between the Z-score of SI and all other parameters. A strong correlation was observed with several OF parameters, such as speed ($R = -0.56$, $p = 1.76e-5$), total distance ($R = -0.54$, $p = 4.27e-5$), look-around ($R = 0.48$, $p = 0.0004$), and inner zone: entries ($R = -0.47$, $p = 0.0004$), but not with the SA-ratio ($R = -0.13$, $p = 0.37$). **D** The SA-ratio shows a significant main effect with the Kruskal-Wallis: $H(2) = 21.22$, $p < 0.0001$. Wilcoxon posthoc shows that SUS animals (SI-ratio < 1) have a significantly lower SI-ratio compared to NS animals $W(9,30) = 249$, $p = 4.1e-5$ and RES animals $W(9,24) = 216$, $p = 1.56e-7$. There is no difference between NS and RES animals $W(30,24) = 270$, $p = 0.12$. **E** The PCA for SA shows a significant main effect with the one-way ANOVA: $F(2,60) = 10.90$, $p = 9.19e-5$. **F** The PC1 eigenvalues of the SA show a significant difference between SUS and NS animals Post-hoc Benjamini Hochberg (BH): $T(9,30) = p = 0.0005$ and between SUS and RES animals $T(9,24) = p = 5.88e-5$. There is no significant difference between NS and RES animals $T(30,24) = p = 0.196$. **G** The PCA for the 2.5 min SI ratio shows a significant main effect with the Kruskal-Wallis: $H(2) = 24.83$, $p = 4.06e-6$. **H** The PC1 eigenvalues of the 2.5 min bin SI show a significant difference between NS and RES animals Post-hoc Wilcoxon: $W(30,24) = 92$, $p = 1.82e-6$, and between NS and SUS animals $W(30,9) = 41$, $p = 0.0015$. There is no difference between RES and SUS animals ($W(24,9) = 117$, $p = 0.736$). **I** B-look-around shows a significant main effect with the one-way-ANOVA: $F(2,60) = 19.23$, $p = 3.53e-7$. Post hoc BH shows a significant difference between NS and RES ($T(30,24) = p = 9.86e-7$), and NS and SUS ($T(30,9) = p = 0.0002$), but no difference between RES and SUS ($T(24,9) = p = 0.94$).

J B-huddle shows a significant main effect with the one-way-ANOVA: $F(2,60) = 12.35$, $p = 3.23e-5$. Post hoc BH shows a significant difference between NS and RES ($T(30,24) = p = 0.0003$), and NS and SUS ($T(30,9) = p = 0.0004$), but no difference between RES and SUS ($T(24,9) = p = 0.39$). **K** B-speed shows a significant main effect with the one-way-ANOVA: $F(2,60) = 18.63$, $p = 5.1e-7$. Post hoc BH shows a significant difference between NS and RES ($T(30,24) = p = 3.12e-6$), and NS and SUS ($T(30,9) = p = 7.62e-5$), but no difference between RES and SUS ($T(24,9) = p = 0.67$). **L** B-nose-to-tail shows a significant main effect with the Kruskal-Wallis: $H(2) = 26.70$, $p = 1.59e-6$. Post hoc Wilcoxon shows a significant difference between NS and RES ($W(30,24) = 628$, $p = 1.82e-6$), and NS and SUS ($W(30,9) = 236$, $p = 0.0005$), but no difference between RES and SUS ($W(24,9) = 152.5$, $p = 0.075$). **M** B-nose-to-body shows a significant main effect with the Kruskal-Wallis: $H(2) = 19.61$, $p = 5.52e-5$. Post hoc Wilcoxon analysis shows a significant difference between NS and RES ($W(30,24) = 567$, $p = 0.0003$), and NS and SUS ($W(30,9) = 230$, $p = 0.0009$), and RES and SUS ($W(24,9) = 167$, $p = 0.018$). The correlation analyses (A, B) are represented with a regression line and a 95% confidence interval window and all individual samples as points. **C** has the correlation value (R) represented by the red line (positive) or blue line (negative), black circles around the points are identified as significant correlations, $p < 0.05$. The bar graphs are presented as mean \pm standard error of the mean and all individual samples as points. The PCA graphs (E, G) are provided with a 95% confidence ellipse and all individual samples as points. Further PCA analyses are represented with a violin plot and all individual samples as points (F, H). The bar graphs are presented as mean \pm standard error of the mean and all individual samples as points. $N = 30$ for NS and CSDS in (A), and $n = 26$ for NS and $n = 27$ for CSDS in (B, C), $n = 30$ for NS, $n = 24$ for RES, $n = 9$ for SUS in (D–M). Source data are provided as a Source Data file.

The expression between NS and CSDS animals was then compared using 2-way Mann-Whitney U tests for each cluster independently, and p values were corrected for multiple testing using the BH method across both clusters and time bins, when applicable. We observed significant differences in eight out of ten and six out of ten clusters for the first time bin of the single and multi-animal SI settings, respectively (Fig. 6E and Supplementary Fig. 5E). Interestingly, and in line with habituation to the environment, these differences also fade across time. The single-animal setting still shows some (although less) significant differences in all time bins, albeit with reduced effect sizes (Supplementary Fig. 10). Interestingly, also in the single-animal embeddings, cluster 8 remains highly significant during the entire course of the experiments. The multi-animal setting yields in contrast almost no significant results beyond the first time bin (Supplementary Fig. 11).

In the OF setting, 7 out of 11 clusters showed a significant differential expression in the first 595 s (Supplementary Fig. 6E). The SA test, in turn, is an interesting setting to test DeepOF given that its main axis of variation is the distance to the cage with the conspecific, which constitutes information that is not available to DeepOF in its current form (which only looks at the posture of the tracked animals). Interestingly, and while the analysis shows no significant results in trial one (without the conspecific, Supplementary Fig. 7E), 6 out of 17 clusters show significant differential expression in trial two (with the conspecific, Supplementary Fig. 7F), suggesting that DeepOF can correctly detect behavioral differences even without absolute location information.

Finally, we also explored the spatial distribution of cluster expression across all three settings. We obtained heatmaps depicting the global exploration of the arena by the C57Bl/6N across all videos (for both conditions). Along these lines, our results show how, while, as shown, CSDS animals tend to occupy the center of the arena significantly less (Fig. 2F) there is no spatial preference across animals for individual clusters (Fig. 6F and Supplementary Figs. 5F, 6F show the overall locomotion distribution, while a comprehensive overview of individual clusters is presented in Supplementary Figs. 12, 13, and 14).

Individual unsupervised clusters reveal differences in behavior dynamics

Aside from comparing cluster enrichment, DeepOF can help gain insight into how cluster transitions and sequences differ across conditions. To accomplish this, an empirical transition matrix was obtained for each condition by counting how many times an animal goes from one given cluster to another (including itself). Since all transitions were observed to have non-zero probability, the Markov chains obtained from simulations can be proven to reach a steady state over time (where probabilities to go from one behavior to another stabilize). The entropy of these steady state distributions was reported for both conditions, with higher values corresponding to a less predictable exploration of the behavioral space. Interestingly, CSDS animals showed a significantly lower behavioral entropy in the social interaction task than their NS counterparts, retrievable in both single and multi-animal embeddings (Fig. 6F and Supplementary Fig. 5F). This goes in line with the NS animals exploring the behavioral space more thoroughly, while CSDS animals are more conditioned by the conspecific. In line with this hypothesis, no significant differences across conditions were found in the single-animal OF experiments (Supplementary Fig. 6F). Moreover, to validate these results, the obtained behavioral entropy score was correlated with the physiology Z-score presented earlier (Supplementary Fig. 15). As expected, significant negative correlations were found for the SI setting both when exploring the single and multi-animal behavioral spaces. No significant correlation was observed for the single-animal OF setting.

Shapley additive explanations reveal a consistent profile across differentially expressed clusters

An important aspect of any machine learning pipeline using highly complex models is its explainability. In this study, we aimed to explain cluster assignments by fitting a multi-output supervised classifier (a gradient boosting machine) that maps statistics of the initial time series segments (including locomotion and individual body part areas, speeds, distances, and angles) to the subsequent cluster assignments. Performance and generalizability of the constructed classifiers across the dataset were assessed in terms of the balanced accuracy on a 10-

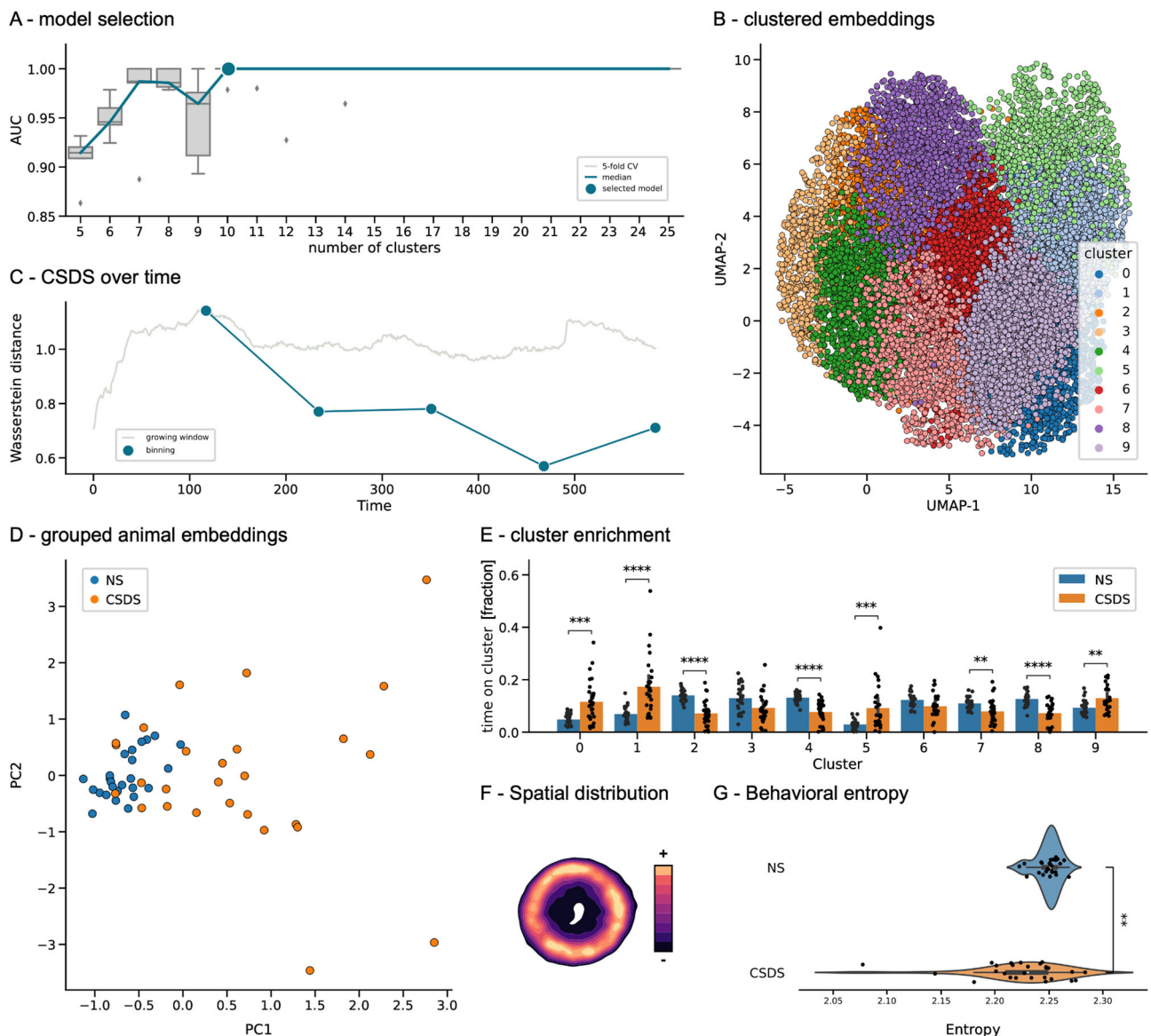


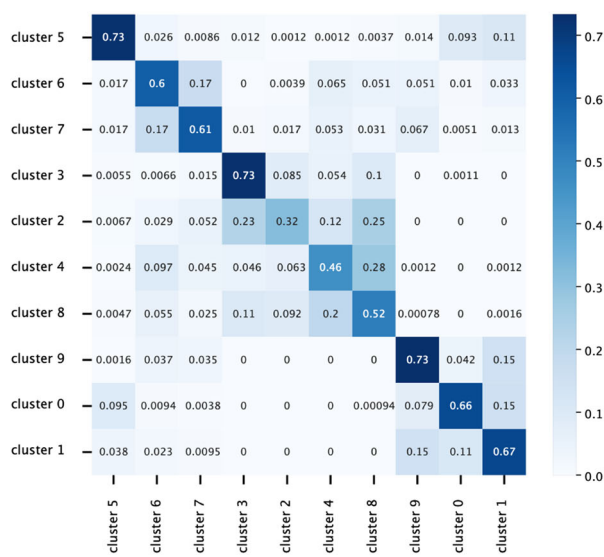
Fig. 6 | Single-animal unsupervised analyses identify different behavioral patterns between stressed and non-stressed mice during the SI task. **A** Cluster selection pipeline results reporting the area under the ROC curve from a logistic regression classifier discriminating between conditions. A 10-component solution (from a range between 5 and 25) was selected as optimal in a fivefold ($N=5$) cross-validation loop (see methods for details). **B** Embeddings by time point obtained using DeepOF's unsupervised pipeline. Different colors correspond to different clusters. Dimensionality was further reduced from the original 8-dimensional embeddings using UMAP for visualization purposes. **C** Optimal binning of the videos was obtained as the Wasserstein distance between the global animal embeddings of both conditions across a growing window, between the first 10–600 s for each video at one-second intervals (gray curve). Higher values correspond to larger behavioral differences across conditions. A maximum was observed at 126 s, close to the stipulated 150 s selected based on the SA task literature. The dark green curve depicts the Wasserstein distance across all subsequent non-overlapping bins with optimal length. The decay observed across time is consistent with the hypothesized arousal period in the CSDS cohort. **D** Representation of the global animal embeddings for the optimally discriminant bin (126 s) per experimental video colored by condition (see methods for details).

E Cluster enrichment per experimental condition ($N=26$ for NS and $N=27$ for CSDS) in the first optimal bin (first 126 s). Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using Benjamini-Hochberg's method across both clusters and bins (significant differences observed in clusters 0: $U=1.6e+2$, $p=7.7e-4$, 1: $U=1.1e+2$, $p=1.3e-5$, 2: $U=6.3e+2$, $p=1.1e-6$, 4: $U=6.4e+2$, $p=3.3e-7$, 5: $U=1.6e+2$, $p=6.3e-4$, 7: $U=5.3e+2$, $p=1.3e-3$, 8: $U=6.2e+2$, $p=1.9e-6$, 9: $U=1.9e+2$, $p=4.4e-3$). Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. **F** Example heatmap depicting spatial distribution across all experiments (in both conditions) for all clusters. Specific heatmaps for all individual clusters are available in Supplementary Fig. 12). **G** Behavioral entropy scores per condition. NS animals show a significantly higher entropy than CSDS animals, which can be attributed to a less predictable exploration of the behavioral space ($U=5.3e+2$, $p=1.68e-3$, $N=26$ for NS and $N=27$ for CSDS). Moreover, and in accordance with these results, behavioral entropy shows a significant negative correlation with the presented stress physiology Z-score (Supplementary Fig. 15A). Source data are provided as a Source Data file. Box plots in (A, G) show the median and the inter-quartile range. Whiskers show the full range, excluding outliers as a function of the inter-quartile range.

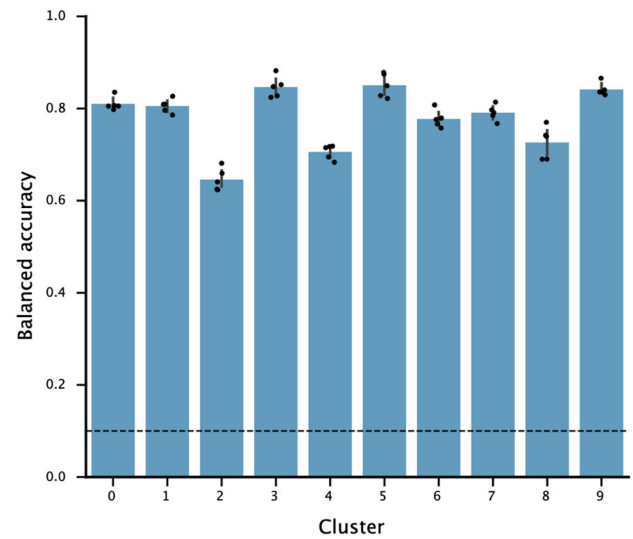
fold stratified cross-validation loop, which was designed so that segments coming from the same video were never assigned to both train and test folds. Data for SI (single and multi-animal) and OF settings were standardized, and the minority class was oversampled using the

SMOTE algorithm to correct for class imbalance. Performance per cluster is shown by means of the confusion matrices per task and the balanced accuracy per cluster (Fig. 7A, B and Supplementary Figs. 16A, B and 17A, B for all three settings, respectively). Importantly, classifier

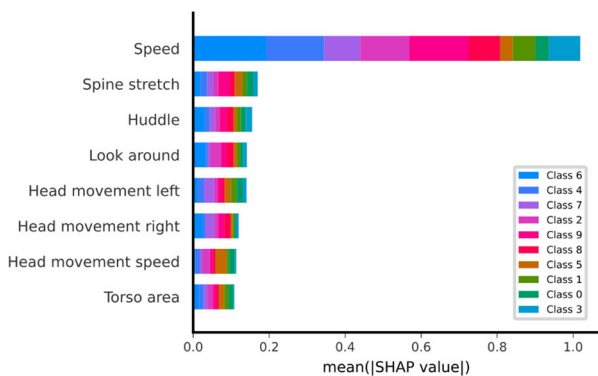
A - cluster detection confusion matrix



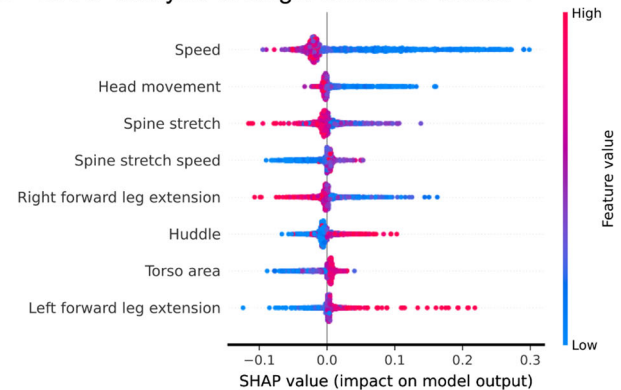
B - cluster detection performance



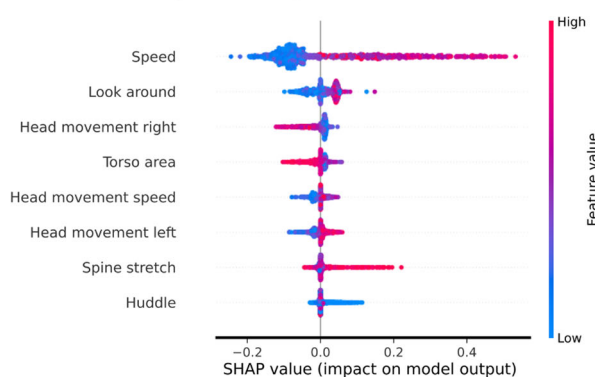
C - SHAP global feature importance



D - SHAP analysis of single-animal SI cluster 1



E - SHAP analysis of single animal SI cluster 2



F - SHAP analysis of single animal SI cluster 8

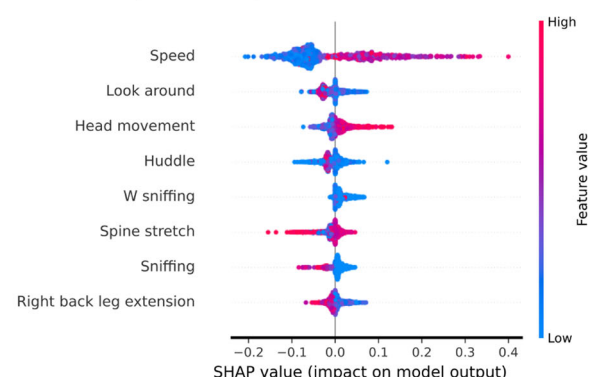


Fig. 7 | SHAP analysis of unsupervised cluster assignments in the single-animal social interaction task. Gradient boosting machines were trained to map from a predefined set of time series statistics (including body part speeds, distances, distance speeds, areas, area speeds, and supervised annotations) to the previously obtained cluster assignments. **A** Confusion matrix obtained from the trained gradient boosting machine classifying between clusters. Aggregated performance over the validation folds of a fivefold cross-validation is shown. **B** Validation performance per cluster across a fivefold ($N = 5$) cross-validation loop. Balanced accuracy was used to correct for cluster assignment imbalance. The dashed line marks the expected performance due to chance, considering all outputs. Bars show

mean \pm 95% confidence interval. **C** Overall feature importance for the multi-output classifier using SHAP. Features in the y-axis are sorted by overall absolute SHAP values across clusters. Classes on the bars are sorted by overall absolute SHAP values across features. **D–F** Bee swarm plots for the three most differentially expressed clusters between NS and CSDS mice (1, 2, and 5), identified with the unsupervised DeepOF pipeline on the SI experiments using single-animal embeddings. The depicted plots display the first eight most important features for each classifier, in terms of the mean absolute value of the SHAP values. Source data are provided as a Source Data file.

performance is substantially greater than random in all cases for all three settings, meaning that all clusters are highly distinguishable from one another by the set of summary statistics employed.

The result of this analysis is thus a set of feature explainers for each retrieved cluster, which can be used to interpret, alongside visual inspection of the corresponding video fragments (included as Supplementary files), what the obtained behavioral motifs represent. Both global (Fig. 7C, Supplementary Figs. 16C, 17C) and cluster-specific feature importance values can be retrieved. In this context, we found consistent descriptions of clusters that are differentially represented across conditions for all three tasks.

In the single-animal SI task, for example, cluster 1 (Fig. 7D, enriched in CSDS animals) is consistently explained by low locomotion speed, low head movement, and low spine stretch, and is positively associated with the huddle classifier. Visual inspection reveals a behavior close to freezing. Cluster 2 (Fig. 7E, enriched in NS animals) is in contrast explained by high locomotion speed, exploratory behavior, low head movement, and spine stretch. Close visual inspection depicts active locomotion and engagement with the conspecific. Interestingly, cluster 8 (Fig. 7F, enriched in NS animals across all time bins) is explained by increased speed, head movement, and negatively associated with sniffing. Visual inspection suggests engaging in motion (shifting from a still position to active locomotion).

In the case of the multi-animal SI setting, the explainability pipeline reveals how the models work differently when taking both animals into account. In this case, the two-animal system is embedded as a whole, and features including both animals are considered when running SHAP. As mentioned in the methods section, a regularization hyperparameter allows the system to focus more on interactions between the animals or in joint individual behaviors. In this case, we used a moderated value of the parameter that enables the contribution of both, which becomes apparent when analyzing the explainability profiles of the retrieved behaviors. Cluster 3, for example (Supplementary Fig. 16D, highly enriched in CSDS), is explained not only by low speed on the C57Bl/6 N animal, but also by increased speed of the CD1, among others. Upon visual inspection, one can observe exactly that the CD1 is exploring the arena while the C57Bl/6N stands still, in a posture usually associated with the stopped and huddled trait. Cluster 5 (Supplementary Fig. 16E, also enriched in CSDS) closely captures an interaction between the two animals, where the CD1 is typically more engaged in movement. The SHAP pipeline eloquently reveals negative correlations with spine stretch and back, torso, body and head areas, as well as speed in both mice. Conversely, cluster 8 (Supplementary Fig. 16F, enriched in NS) is well explained by increased speed in both animals, which can be confirmed by visual inspection.

Finally, this pipeline was also used to interpret clusters in the OF setting. In this case, cluster 0 (Supplementary Fig. 17D, enriched in CSDS animals) is explained by a decreased overall speed, positive correlations with mid and back spine stretch, back area, and left leg extensions, and negative association with right leg extensions. Visual inspection indeed reveals a cluster highly enriched in digging. Cluster 8 (Supplementary Fig. 17E, also enriched in CSDS animals), is in turn explained by decreased speed, mid, and back spine stretch, increased head area and extended right legs. Visual inspection shows a cluster enriched in slow walking, often including head movement and interaction with the walls. Finally, cluster 9 (Supplementary Fig. 17F, enriched in NS animals) is positively correlated with speed and head movement, and negatively correlated with spine stretch, among others. Visual inspection depicts an exploratory behavior with active movement.

All in all, the provided cluster explainability pipeline is a useful tool to interpret all reported patterns. Moreover, visual inspection of cluster snippets is also made possible with a single command within DeepOF, which makes the interpretation process more effective.

Discussion

For decades there has been a trend to standardize and simplify social behavioral tests, which has led to an oversimplification of the description of the social behavioral repertoire. The current developments of open-source markerless pose estimation tools for tracking multiple animals have provided the possibility for more complex and socially relevant behavioral tests. The current study provides an open-source tool, DeepOF, which can investigate both the individual and social behavioral profiles in mice using DeepLabCut-annotated pose estimation data. Applying this tool, the current study identified a distinct social behavioral profile following CSDS using a selection of five traits annotated by DeepOF on the C57Bl/6N animal. In addition, a similar social behavioral profile was identified using an unsupervised workflow, which could detect behavioral differences in different experimental settings, including social interaction and single-animal open field tests, and a social avoidance task. Moreover, DeepOF allowed to study behavioral dynamics in unprecedented detail and identified the 5 min during the interaction with a novel conspecific as crucial for the social profiling of CSDS exposure in both supervised and unsupervised workflows. Overall, this study demonstrates the high utility and versatility of DeepOF for the analysis of complex individual and social behavior in rodents.

DeepOF as part of a markerless pose estimation toolset

The initial release of DeepLabCut in 2018²⁹ provided a reliable and accessible tool for researchers around the globe to process markerless pose estimation data, which has undoubtedly changed the field of behavioral neuroscience. This has set in motion a rapid growth of tools for analyzing pose estimation data that are increasing the range of possibilities in the field, which were unimaginable using classical tracking approaches or manual scoring. An important distinction between these pose estimation analysis tools is whether they intend to extract pre-defined and characterized traits (supervised) or to explore the data and extract patterns without external information (unsupervised). The DeepOF module is designed to provide both analysis pipelines. The supervised behavioral classifiers offer a quick and easy-to-use analysis to detect individual and social behavioral traits without manual labeling. In addition, when differences between the conditions are not reflected in these traits, or the researcher aims to obtain behavioral embeddings, the DeepOF package can encode the data in a time-aware way that can report differentially expressed patterns in an unsupervised manner, taking single and multi-animal inputs.

The supervised framework: spotting recognizable patterns

The supervised pipeline within the DeepOF package can be used on single and dyadic behavioral data in multiple-shaped arenas. DeepOF is capable of reporting a pre-defined set of behavioral traits without any extra labeling or training required. To accomplish this, it relies on both simple rule-based annotations and machine learning binary classifiers whose generalizability has been tested, trading off flexibility for ease of use. This makes it user-friendly for researchers without computational expertise to apply this supervised pipeline, without having to make any modifications. To further detect unsupported patterns, using a more involved and flexible tool (such as SimBA³⁷ or MARS²⁷) could be a reasonable next step to take. These tools include a supervised approach that requires the user to label and train classifiers, providing the freedom to train powerful classifiers and recognize behavioral traits, which is especially beneficial for labs without computational expertise. However, in contrast to DeepOF, this approach also delegates to the user the responsibility of testing the generalizability of the results (how well the trained models can be applied to newly generated data, even in similar settings), which requires careful practices from the experimenters.

The DeepOF module provides a more complete social behavioral profile than the social avoidance task

The social behavioral profile in CSDS-subjected animals has been measured extensively using the SA task, which is based on the separation of social behavioral traits between non-stressed and stressed animals^{11,17,38}. Previous research has shown that rodents have a social interaction preference towards a novel conspecific compared to a familiar conspecific³⁹. However, the duration of this social behavioral arousal state has not been well documented. In this context, and by replicating the time the SA task typically lasts for¹⁰, the current study shows that the CSDS-related social behavioral profile, obtained with the DeepOF supervised classifiers, was increasingly observed during the first 2.5 min of the 10 min SI task. Furthermore, the presented unsupervised workflow was used to determine an optimal binning of our experiments by measuring how different both conditions were across time for a linear classifier. This yielded an optimal separation at -2.1 min (126 and 124 s when testing with single and multi-animal embeddings, respectively), which then decayed over subsequent time bins in a manner consistent with the arousal hypothesis. The fact that this result was not seen in the absence of a conspecific strengthens this argument. Taking this into account, we argue that the introduction of a novel conspecific induces a state of arousal, which coincides with a distinct social behavioral profile that disappears over time after 2–3 min due to habituation.

Along these lines, this study shows that the DeepOF social behavioral classifiers provide a stronger separation of the social behavioral profile between stressed and non-stressed animals compared to the classical SA task, which also correlates better to physiological stress parameters.

Furthermore, the identification of stress-susceptible and resilient animals is often performed using the SA-ratio of the SA task^{10,17} and for this DeepOF offers unique advantages. While the SA ratio clearly distinguishes stress-affected individuals, especially following more severe CSDS paradigms, the DeepOF module will significantly advance the possibilities and sensitivity of this distinction, by investigating the degree of resilience based on multiple behavioral classifiers with high sensitivity and in freely moving animals, which enables uncovering a so far undescribed set of resilience-linked phenotypes that are different from the univariate SA task. Taken together, it can be concluded that using the DeepOF social behavioral classifiers provides a more robust and clearer social behavioral profile in animals subjected to CSDS compared to the SA task. An important reason for the superiority of DeepOF in social behavioral profiling depends on the experimental setup: the SA task relies on the confinement of an animal (for example using a wired mesh cage), which means that no natural interaction between freely moving animals is possible, whereas the SI task is based on a naturally occurring interaction between freely moving animals¹⁸. Moreover, in the SA task, the confined animal can show symptoms of anxiety-related behavior, which influences the physiological state and the social interaction and approach behavior of the conspecific^{40–42}. Differences in anxiety-related behavior between experimental animals can still contribute to alterations in social behavior and recent data suggest distinct neurobiological circuits driving both phenotypes⁴³, therefore sufficient habituation and the ability to observe behavior in freely moving animals will lead to improved discrimination. Moreover, a further crucial advantage of the DeepOF module is the many different behavioral classifiers that can be investigated at the same time without increasing the labor intensity. The combined analysis of multiple behavioral classifiers into a Z-score of social behavior provides a more complete social behavioral profile than solely investigating social avoidance behavior.

DeepOF can detect and explain differences across experimental conditions in a fully unsupervised way, embedding data from one or more animals

The supervised pipeline within DeepOF follows a highly opinionated philosophy, which focuses on ease of use and relies on predefined

models. As an alternative, DeepOF offers an unsupervised workflow capable of representing animal behavior across experiments without any label information. In its most basic expression, this involves obtaining a representation for each experiment in a time-aware manner: unlike other dimensionality reduction algorithms like PCA, UMAP, and T-SNE²⁶, DeepOF, when applied to the raw dataset, relies on a combination of convolutional and recurrent neural networks capable of modeling the sequential nature of motion. Each input to the models consists of a subsequence across a non-overlapping sliding window of each experiment. Although this idea has been explored before³³, DeepOF introduces several novelties to the field, such as unified embedding and clustering, the support for multi-animal embeddings, and graph representations that integrate not only coordinates by also body-part-specific speed and distance information, which makes it ideal for settings where informative body parts (such as paws) are occluded, as is the case for commonly used top-down videos.

In addition, these global embeddings can be decomposed into a set of clusters representing behavioral motifs that the user can then inspect both visually and with machine learning explainability methods. Moreover, by comparing cluster enrichment and dynamics across conditions, it is possible to answer questions that are relevant to understanding what the observed difference might be based on, without any previous knowledge: Which behaviors are most or least expressed in each condition? Is the set of behaviors expressed differently in experimental conditions? Are they expressed differentially across space and time? This constitutes a complementary approach that can be beneficial to further direct hypotheses when little knowledge is available. In addition, by not only showing overall differences between cohorts but also reporting which motion primitives might be driving them, it is possible to test hypotheses by training novel supervised classifiers based on those motion primitives. This can allow researchers to distinguish new, meaningful patterns that have not been reported before and that may be significantly associated with a given condition.

Taken together, the current study exemplifies that the unsupervised pipeline provided in DeepOF does not only recapitulate results previously obtained with the supervised analysis, but also shows how this tool can be used to detect habituation and overall differences in behavioral exploration. We also show that detected differences are significantly stronger when a conspecific is present, although also detectable during single animal arena exploration alone.

Towards an open-source behavioral analysis ecosystem

One of the main advantages of DeepOF, SimBA³⁷, VAME³³, MARS²⁷, and many other packages cited in this manuscript, is that they are open source. This means that their inner workings are transparent, and that it is possible for the community to contribute to their development. We strongly believe that the adoption of open-source frameworks can not only increase transparency in the field but also incentivize a feeling of community, in which researchers and developers can share ideas, code, and solve bugs and problems together. Moreover, the open source framework facilitates beneficial feedback loops, where the data generated using these tools can be published, thus increasing the opportunity to produce better software. A good example of this is zero-shot pose estimation⁴⁴, which enables motion tracking without labeling, by cleverly leveraging information from several publicly available datasets. In addition, new technologies are starting to enable joint learning from multiple modalities, such as neural activation and behavior⁴⁵, which enables the exploration of how these modalities are influencing each other.

In addition to the software, an equally important problem to tackle is the need for open-source benchmarks. As platforms for testing and validating pose estimation and detection algorithms become available, it becomes easier to clearly show and compare the performance of different software options for different tasks. An

example of this is the Caltech Mouse Social Interactions (CalMS21) dataset, a pioneer in the field that provides benchmarking for classic detection of social interactions, annotation style transfer, and detection of rare traits⁴⁶. While unsupervised learning benchmarking remains highly unexplored to the best of our knowledge, it would be crucial to compare the DeepOF pipeline with other available methods in this context when the tools become available.

Finally, and in contrast to several other options that offer extended functionality but rely on proprietary algorithms and/or specialized hardware²³, these tools have the potential to make otherwise expensive software available to a larger audience.

In conclusion, the current study provides a novel approach for individual and social behavioral profiling in rodents by extracting pre-defined behavioral classifiers and unsupervised, time-aware embeddings using DeepOF. Furthermore, while the tool provides means of customization, it is uniquely optimized for the most common behavioral setup: top-down video recordings. Moreover, we show evidence for the validation of the provided behavioral annotators and offer an open-source package to increase transparency and contribute to the further standardization of the behavioral constructs. We also show that, while differences across conditions are detectable during single animal exploration, they are enhanced in the SI task involving a companion mouse. Furthermore, while the classical SA task does identify the social behavioral profile induced by CSDS, the DeepOF behavioral classifiers provide a more robust and clearer profile. DeepOF is thereby a highly versatile tool that can also be applied to other research questions, e.g., to study sex differences in social behavior or analyze home-cage behavior throughout the lifespan of animals using longitudinal recordings. In addition, the DeepOF module contributes to a more specific classification of the affected individual and social behaviors in stress-related disorders, which could contribute to the study of drug development for psychiatric disorders.

Methods

Time series extraction from raw videos

Time series were extracted from videos using DeepLabCut version 2.2b7 (single animal mode). 11 body parts per animal were tagged, including the nose, left and right ears, three points along the spine (including the center of the animal), all four extremities, and the tail base (Fig. 1A). The DeepLabCut model was trained to track up to two animals at once (one CD1 mouse and one C57Bl/6 N mouse) and can be found in the Supplementary material (see code and data availability statement). Using the multi-animal DeepLabCut³⁰, extending the tracking to animals from the same strain is also possible. Next, DeepLabCut annotated datasets were processed and analyzed using DeepOF v0.4.6³⁶.

Time series data preprocessing

All videos and extracted time series undergo an automatic preprocessing pipeline that is included within the DeepOF package, consisting of smoothing and two sequential imputation levels, applied to all body parts of all tracked animals independently. For smoothing DeepOF applies a Savitzky-Golay filter⁴⁷ to each independent tracked variable by fitting an $n/2$ -degree polynomial over an n -frame sliding window, where n is the frame rate of the corresponding videos.

To identify and correct any artifacts in the time series, a moving average model is then fitted to the time-based quality scores of each tracked variable (as reported by DeepLabCut's output likelihood). By detecting divergences (of at least three standard deviations) from the moving average model, DeepOF can detect sudden and consistent drops in tracking quality, often correlated with body-part occlusions. Body parts with low quality are thus removed from the data, and further imputed using `sci-kit learn`'s iterative imputer with default parameters⁴⁸, which predicts missing values based on all available

features at a given time point using a Bayesian ridge regression method. A second imputation method is then conducted, aiming to remove spatial jumps in the tracked body parts. To do this, another moving average model is fitted, this time to the body part coordinates themselves, and any data point located at least three standard deviations from the model is replaced by the predicted values.

Time series feature extraction

After preprocessing the time series independently, DeepOF extracts a set of features aiming to describe how entire animals move and interact. These include centered and aligned coordinates, distances between body parts, angles, and areas of specific regions of each available body (Fig. 1B), as well as their speeds, accelerations, and higher-order derivatives. The value for each feature is reported per time point.

Coordinates. Raw coordinates for each body part are centered (the cartesian origin is set to the center of each animal) and vertically aligned so that the y -axis matches with the line delimited by the *center* of each animal and *spine 1* (see Fig. 1A for reference). This is done so that both translational and rotational variances are not considered in further processing steps (in principle, and except for some annotations such as wall climbing and sniffing—see below—DeepOF extracts posture patterns that are invariant to where in the arena and in which rotational orientation they are expressed).

Distances and angles. Distances and angles over time between all body parts within and across all animals are computed by DeepOF by default, and available for retrieval.

Areas. The full area of the animal over time is computed by DeepOF by defining a polygon on all external body parts (*nose*, *ears*, *legs*, and *tail base*). The head area is delimited by the *nose*, *ears*, and *spine 1*. The Torso area is delimited by *spine 1*, both *forward legs*, and *spine 2*. The back area is delimited by the *center*, both *back legs*, and the *tail base*.

Finally, speeds, accelerations, jerks, and larger-order derivatives of each extracted feature are also computed using a sliding window approach. Importantly, the detailed 11-body-part labeling scheme suggested and provided by DeepOF plays a crucial role here. While parts of the pipeline can still work with fewer labels, the comprehensive set of features that DeepOF is able to extract with this set of labels enhances not only supervised annotations, but also data representations and model interpretability.

Supervised behavioral tagging with DeepOF

The supervised pipeline within DeepOF aims to provide a set of annotators that work out of the box (without user labeling) for several behaviorally relevant traits. The workflow supports both dyadic interactions and individual traits, which are reported for each mouse individually (Fig. 1C). Furthermore, annotated traits fall into one of two categories:

1. *Traits annotated based on pre-defined rules.* Several motifs of interest are annotated using a set of rules that do not require a trained model. For example, contact between animals can be reported when the distance between the involved body parts is less than a certain threshold.
2. *Traits annotated following a supervised machine learning pipeline.* While rule-based annotation is enough for some traits, others are too complex or might be manifested in subtly different ways, and machine learning models are often a better option. In this case, a rigorous validation pipeline has been applied to measure the performance of the classifier not only in a separate test data set, but also across datasets comprehending different arenas and laboratories.

Rule-based annotated traits. Among the rule-based annotated dyadic traits, nose-to-nose and nose-to-tail depend on single distance thresholds between specific body parts of the animals involved. In the case of nose-to-body, a single threshold is used between the nose of one animal and any body part of the other (except nose and tail base). Side-by-side and side-reverse-side are computed using two equal thresholds, measuring the distance between both noses and two tails in the former, and both nose-to-tail distances in the latter.

Of the individual traits, “look around” requires the animal to stand still (speed to be below a defined threshold) and the head to be moving (nose and ear speeds to be above a defined threshold). Finally, sniffing and wall climbing rely on the interaction of each animal with the arena (which can be detected automatically in certain settings, or indicated manually by the user using a GUI—graphical user interface—when creating a DeepOF project). An animal is annotated as sniffing the walls when speed is below a defined threshold, the distance between the nose and the wall is below a defined threshold, and the head is moving. Consequently, wall climbing is detected when the nose of an animal goes more than a certain threshold beyond the delimited arena. All mentioned thresholds can be specified (in millimeters) by the user. All analyses presented in this article were conducted with default values, which can be seen in Supplementary Table 1. Moreover, all annotations require a reported tracking likelihood of at least 0.85 on all involved body parts.

Annotation using pre-trained machine learning models. In the case of stopped and huddled, we trained a gradient boosting machine (scikit-learn, v1.2.0, default parameters) to detect the trait per frame, using a set of 26 variables including distances between body parts, speeds, and areas. Data were preprocessed by standardizing each animal’s trajectories independently (controlling for body size), and the training set as a whole. Furthermore, to deal with the imbalanced nature of the dataset (as only 8.48 % of the frames were positively labeled) we applied Synthetic Minority Over-sampling Technique (SMOTE)⁴⁹ to oversample the minority class (using imblearn v0.10.1⁵⁰).

Performance was then evaluated using a tenfold stratified cross-validation (to keep approximately the same number of positive labels in each validation fold) on a single dataset for model development and tested externally using a leave-one-dataset-out approach. Four independent datasets were used, collected in four different settings and across two different labs (see dataset details in Supplementary Table 2). Three of them (SI, OF, and SA) were tagged with manual labeling only, whereas the fourth (EX, obtained externally) combined manual labels and automatic pseudo-labeling using SimBA (Supplementary Fig. 2). The final classifier deployed with the latest version of DeepOF was then trained on a set of more than half a million labeled frames (567,367), coming from all four mentioned independent datasets, and global feature importance was obtained using SHAP (Shapley additive explanations).

After applying the annotators, a Kleinberg burst detection algorithm^{37,51} is applied to all predictions. This step smoothens the results by merging detections that are close in time (called bursts) and removing isolated predictions, which an infinite hidden Markov model deems as noise. Moreover, rather than having a fixed detection window, the filter will be less likely to ignore isolated or less frequent events if they are far enough from higher frequency bursts but will be more prone to removing isolated events closer to a region where annotations are more frequent. In addition, it is important to notice that the annotators work independently, so more than one label can be assigned to an animal at a given time point (Fig. 1D).

Overall, while the provided behavioral set may not cover all scenarios, this out-of-the-box pipeline can be used to detect differences in behavior across experimental conditions without the need for further programming. More complex behaviors, involving user definition and labeling can thus be extracted using other available tools if required³⁷.

Graph representations

To analyze complex spatio-temporal data involving features such as coordinates, speed, and distances, the unsupervised pipeline within DeepOF can structure the variables as an annotated graph (Fig. 1E).

In this representation, each node is annotated with three values, corresponding to both coordinates of each body part, as well as their speeds. Edges are in turn annotated with distances between both connected body parts. The adjacency matrix describing connectivity is provided by DeepOF for top-down videos, but can also be defined by the user. Moreover, this representation can be extended to a multi-animal setting, where independent graph representations for each animal are connected through nose-to-nose, nose-to-tail, and tail-to-tail edges, allowing the models to incorporate relative distances between animals. It is worth mentioning that the provided representation works best when adjacent body parts are being tracked so that propagation through space is not too coarse. One of the main assumptions behind spatio-temporal graph embeddings is that connected body parts are sufficiently correlated in space, which may not be the case if too little tracking labels are included⁵².

Unsupervised deep embeddings with DeepOF

Unsupervised analysis of behavior was conducted using an integrated workflow within DeepOF, which enables both the deep embedding of animal trajectories and their clustering, to retrieve motion motifs that are consistent across time.

To this end, node and edge features (for either single or multiple animals) are processed using a sliding window across time, and standardized twice: once per animal, to remove size variability, and a second time on the entire training set.

The resulting data is then embedded using a deep clustering neural network architecture based on Variational Deep Embeddings^{53,54}, a deep clustering algorithm that can be adapted to sequential data. During training of the models, DeepOF minimizes the ELBO (evidence lower bound), represented in Eq. (1):

$$L_{\text{ELBO}}(x) = \mathbb{E}_{q(z,c|x)}[\log p(x|z)] - D_{\text{KL}}(q(z,c|x)||p(z,c)) \quad (1)$$

The first term corresponds to the reconstruction loss, which encourages the latent space (z) to represent the data (x) well over a set of clusters (c). The second term is the Kullback-Leibler divergence (D_{KL}) between a mixture-of-Gaussians prior ($p(z,c)$) and the variational posterior for each cluster ($q(z,c|x)$), which regularizes the embeddings to follow a mixture-of-Gaussians distribution where each component is associated with a particular behavior. A schematic overview of the model can be found in Fig. 1F.

Importantly, this loss function enforces a clustering structure directly in the latent space, removing the need for post-hoc clustering of the embeddings required by other available tools³³. This has several advantages, the main one being that the clustering structure back-propagates to the encoder during training, improving clustering performance⁵⁵.

The main contribution of the provided architecture lies however in the encoder-decoder layers, which are designed to handle spatio-temporal graph data (in which connectivity is static, but node and edge attributes change over time)⁵⁶. To accomplish this, features corresponding to each body part are first processed independently by a temporal block, which consists of a one-dimensional convolutional neural network (CNN) and two gated recurrent unit (GRU) layers. Subsequently, the outputs of these layers are passed by a spatial block, that shares information across adjacent body parts. This is accomplished using CensNet convolutions, a graph convolution architecture capable of embedding node and edge attributes at the same time⁵⁷. This allows DeepOF to take advantage of several data modalities related to motion with a single data structure as input.

Once the models are trained, cluster assignments are obtained as the argmax of the posterior distribution given the data, as described in Eq. (2):

$$q(c|x) = p(c|z) \equiv \frac{p(z)p(z|c)}{\sum_{c'=1}^K p(c')p(z|c')} \quad (2)$$

where $c' \in (1, K)$ is an iterator over all clusters in the model.

In practice, this unsupervised pipeline can retrieve consistent patterns of animal motion in a flexible, non-linear, and fully unsupervised way. Moreover, as body part speeds and distances can be naturally included, this workflow works even when critical body parts (such as the paws) are occluded, which makes it ideal for top-down videos.

In addition, DeepOF is capable of training multi-animal embeddings by using multi-animal graphs (see graph representations section above). When more than one animal is detected, DeepOF allows the user to control how much these embeddings should consider interactions between the animals over the multi-animal system. This is achieved with an L1 penalization over the node embeddings in the aforementioned CensNet layers: larger values will prime the models to prioritize animal interactions, whereas smaller values will increase the contribution of the individual behavior of each animal. All experiments included in this study used a moderated parameter (0.25) which allowed the model to consider both interactions and joint individual behaviors.

Unsupervised model training and hyperparameters

All unsupervised models used default values (as specified in DeepOF version 0.4.6). On each dataset, 10% of the available videos were used as a validation set to evaluate performance during training. Data were processed using sliding windows with a length matching the video frame rate of each dataset and stride of 1, mapping to eight-dimensional latent spaces. The training was conducted using the Nadam optimizer⁵⁸ (with a learning rate of 0.001 and gradient-based clipping of 0.75) over 150 epochs with early stopping based on the total validation loss and patience of 15 epochs. Upon training end, weights of the models are restored to those obtained in the best performing epoch using the same metric. The number of populated clusters over time, confidence in selected clusters (as the argmax of the produced soft counts), regularizers, and individual components of the loss function (see unsupervised deep embeddings with DeepOF section above) are tracked over time by DeepOF.

Global animal embeddings

Aside from embedding time points individually, global animal embeddings (where each data point corresponds to the trajectory of an entire animal rather than to a single time point) were obtained by constructing a k -dimensional vector with the time proportion each animal spent on each cluster, where k is the number of clusters in the given model.

Cluster number selection

For each dataset that was analyzed with the unsupervised pipeline, models ranging from 5 to 25 clusters were trained five times, resulting in a total of 120 models per explored setting. All model hyperparameters were set to DeepOF defaults (see section below and API documentation for additional details). Global animal embeddings were then used as input to a logistic regression classifier (scikit-learn, default parameters) aiming to discriminate CSDS from non-stressed animals. The model with the smallest number of clusters that reached a performance within one standard deviation of the global maximum across the whole range (in terms of the area under the ROC—receiver operating characteristic—curve) was selected for further processing.

Time binning and habituation quantification

A key aspect of DeepOF is that it allows for quantification of behavioral differences between cohorts over time in an unsupervised way. In this context, this is done by measuring the Wasserstein distance over time between the multivariate distributions describing global animal embeddings for CSDS and non-stressed animals.

By measuring this distance across a growing window, we can quantify how important additional information is to discriminate between conditions. This way, a peak in the distance curve would mark the point in time in which behavioral differences are maximized. In this study, we used a range between 10 and 600 s for each experiment, computing the Wasserstein distance between conditions every second. The time point at which the maximum was reached was selected as the optimal size for consecutive (non-overlapping) time bins. By reporting the behavioral distance along these bins, DeepOF can report behavioral habituation (which would involve behavioral differences between conditions decreasing over time).

Unsupervised cluster interpretation using Shapley additive explanations (SHAP)

When applying the unsupervised pipeline, and quantifying which features DeepOF deems relevant for the unsupervised models to determine the assignment of a given time segment to a given cluster, all obtained sequence-cluster mappings were analyzed using Shapley additive explanations^{59,60}.

To this end, a comprehensive set of 52 distinct features (111 for two-animal embeddings) was built to describe each sliding window in the training set, including mean values of distances, angles, speeds, and supervised annotators.

Gradient boosting machines (using Catboost v1.11⁶¹, which offers models specifically optimized for non-binary classification) were then trained to predict cluster labels from this set of statistics after normalization across the dataset and oversampling the minority class with the SMOTE algorithm⁴⁹. Performance is reported as the validation balanced accuracy across a 10-fold stratified cross-validation loop, and feature importance (global and for each cluster) is reported in terms of the average absolute SHAP values, obtained using a permutation explainer.

Animals for chronic social defeat stress experiments

Eight-week-old experimental male C57Bl/6N mice were bred in-house. The CD1 male mice (bred in-house) were used in the social avoidance and social interaction task as social conspecifics (CD1 animals were 4–6 weeks old) and as aggressors in the CSDS paradigm (CD1 animals were at least 16 weeks old). The study was conducted with male animals as a proof of principle, and for comparability to widely available data on chronic social defeat. All animals were housed in individually-ventilated cages (IVC; 30 cm × 16 cm × 16 cm connected by a central airflow system: Tecniplast, IVC Green Line—GM500) at least 2 weeks before the start of the experiment to allow acclimatization to the behavioral testing facility. All animals were kept under standard housing conditions; 12 h/12 h light-dark cycle (lights on at 7 a.m.), temperature 23 ± 1 °C, humidity 55%. Food (Altromin 1324, Altromin GmbH, Germany) and water were available *ad libitum*. All experimental procedures were approved by the committee for the Care and Use of Laboratory Animals of the government of Upper Bavaria, Germany. All experiments were in accordance with the European Communities Council Directive 2010/63/EU.

Chronic social defeat stress

At 2 months of age, male mice were randomly divided into the CSDS condition ($n = 30$) or the non-stressed condition (NS) ($n = 30$) (Supplementary Table 2, experiment code 1). The CSDS paradigm consisted of exposing the experimental C57Bl/6 N mouse to an aggressive CD1 mouse for 21 consecutive days, as previously described⁶². An additional

cohort (NS: $n = 30$, CSDS: $n = 33$, subdivided into susceptible animals $n = 9$, and resilient animals $n = 24$) was used to test the DeepOF social interaction classifiers on the resiliency and susceptibility division of the social avoidance ratio (Supplementary Table 2, experiment code 2). The prolonged 3-week CSDS paradigm was specifically chosen to elicit a more profound passive defeat phenotype, as originally reported by Kudryavtseva et al.¹³, and to allow multiple behavioral assessments under stress conditions. In short, the CD1 aggressor mice were trained and specifically selected on their aggression prior to the start of the experiment. The experimental mice were introduced daily to a novel CD1 resident's territory, who attacked and forced the experimental mouse into subordination. Defeat sessions lasted until the stress-exposed mouse received two bouts of attacks from the CD1 aggressor or at 5 min in the rare instances when two bouts were not achieved within this duration. Animal health was monitored throughout the experiment to ensure that any minor injuries healed prior to the subsequent defeat session. Between daily defeats, stressed mice were housed in the resident's home cage but physically separated from the resident by a see-through, perforated mesh barrier, allowing sensory exposure to the CD1 aggressor mouse while preventing further attacks. The defeat time of day was randomized between 11 a.m. and 6 p.m. to avoid habituation and anticipatory behaviors in defeated mice. NS mice were single-housed in the same room as the stressed mice. All animals were handled daily and weighed every 3–4 days. Behavioral testing was performed after 14 days of the defeat paradigm, where behavior was observed in the morning and the defeat continued in the afternoon. The animals were sacrificed a day after the CSDS ended under deep isoflurane anesthesia by decapitation, which was at 3 months of age. Then, the adrenals were obtained, and the relative adrenal weight was calculated by dividing the adrenal weight by the body weight before sacrifice.

Behavioral testing

Behavioral tests were performed between 8 a.m. and 11 a.m. in the same room as the housing facility. On day 15 of the CSDS paradigm, the animals were tested on the social avoidance (SA) task, while on day 16, the animals were tested on the combined open field (OF) and social interaction (SI) task. The SA task was analyzed using the automated video-tracking software AnyMaze 6.33 (Stoelting, Dublin, Ireland), whereas the OF and SI tasks were analyzed using DeepLabCut 2.2b7 for pose estimation^{29,30}, after which DeepOF module version 0.4.6 was used for preprocessing, supervised, and unsupervised analyses of behavior.

Social avoidance

The SA task was performed in a square OF arena (50 × 50 cm) to observe the social behavioral profile after CSDS, as well-established in previous studies^{13,62–64}. The SA task consisted of two phases: the non-social stimulus phase and the social stimulus phase. During the non-social stimulus phase, which was the first 2.5 min, the experimental mouse was allowed to freely explore the OF arena with a small empty wired mesh cage against the wall of the OF. Then, the empty wired mesh cage was replaced with a wired mesh cage including a trapped unfamiliar young CD1 mouse (4–6 weeks old). During the following 2.5 min, the social-stimulus phase, the experimental mouse could freely explore the arena again. The SA-ratio was calculated by calculating the amount of time spent with the social stimulus, which was then divided by the time spent with the non-social stimulus. The identification of CSDS susceptibility and resiliency was obtained using a SA-ratio score of lower than “1” for susceptible animals, and an SI-ratio score higher than “1” for resilient animals.

Open field and social interaction task

The OF and SI tasks were performed in a round OF arena (diameter of 38 cm). The bottom of the arena was covered in sawdust material to

minimize the cross-over effects of stress and anxiety by the novel environment. First, the OF task was performed, during which the experimental animal was allowed to freely explore the arena for 10 min. Subsequently, for the SI task, an unfamiliar young CD1 (4–6 weeks old) was introduced inside the arena and both animals were allowed to freely explore the arena for 10 min. The DeepOF module can identify five behavioral traits during the single animal OF task, which include wall-climbing, stopped-and-huddled, look-around, sniffing, and speed (locomotion), whereas in the SI task, all behavioral traits can be identified (Fig. 1C). During the analysis, the 10 min OF and SI tasks were analyzed in the total duration of the behavioral classifiers, and in time bins of 2.5 min to match the time frame in the SA task.

Z-score stress physiology and social interaction calculation

The Z-scores combine the outcome of multiple tests via mean normalization and provide an overall score for the related behavior of interest. Z-scores were calculated as described previously⁶⁵. The Z-score indicates for every observation (X), the number of standard deviations (σ) above or below the mean of the control group (μ). This means that for each individual observation Eq. (3) is calculated:

$$Z = \frac{X - \mu}{\sigma} \quad (3)$$

Then, the obtained values need to be corrected for the directionality, such that an increased score will reflect the increase of the related behavior of interest. This means that per test, the scores were either already correct or were adjusted in the correct directionality by multiplying with “-1”. Finally, to calculate the final z-score, the different z-scores per test were combined and divided by the total number of tests, as in Eq. (4).

$$Z_{total} = \frac{\sum_i z_{test_i}}{\text{Number of tests}} \quad (4)$$

The Z-score analysis of stress physiology is based on the relative adrenal weight and the body weight at day 21 of the CSDS, which are both strongly influenced by CSDS exposure¹². The directionality of both tests did not require additional adjustment. Then, the Z-score of SI was calculated based on five DeepOF behavioral classifiers from the C57Bl/6N mouse, which were B-look-around, B-speed, B-huddle, B-nose-to-tail, and B-nose-to-body. The directionality was adjusted for B-speed, B-nose-to-tail, and B-nose-to-body.

Behavioral entropy calculation

Shannon's entropy of the behavioral cluster space was obtained directly using DeepOF, as a measure of how predictable the sequence of behaviors expressed by a given animal is. To accomplish this, DeepOF obtains transition matrices across clusters using the unsupervised cluster assignments per animal. Stationary distributions for each transition matrix are then obtained by simulation through the matrices until convergence, and Shannon's entropy is computed for each stationary distribution. Entropy scores obtained for NS and CSDS animals were then compared. Overall entropy scores were also compared to the stress physiology Z-score for validation purposes.

External dataset for validation of the DeepOF huddle classifier

An additional experiment was performed using different conditions and behavioral set-up, to assess the transferability of the DeepOF huddle classifier (Supplemental Table 2, experiment code 3) to data produced by a different lab. 12 weeks old C57Bl/6J mice ($n = 24$, purchased from the Jackson Laboratory (catalog number 000664), Bar Harbor, ME, USA) were paired in a home-cage environment (19 × 19 cm) with 12 weeks old ovariectomized CFW female mice

(purchased from Charles River Laboratories (catalog number 024), Wilmington, MA, USA) and were allowed to freely explore each other for 1.5 min. The animals were housed under standard laboratory conditions with a 12 h light–dark cycle (lights on from 07:00 to 19:00), temperature $22 \pm 1^\circ\text{C}$, humidity 50%, in clear Plexiglas cages ($19 \times 29 \times 13$ cm) with unrestricted access to food (Purina Laboratory Rodent Diet 5001) and water. Procedures were approved by the McLean Hospital Institutional Animal Care and Use Committee and complied with the National Institutes of Health guidelines.

Statistics

Statistical analyses and graphs were made in RStudio (R 4.1.1⁶⁶) and python (v 3.9.13). All data were used for further statistical analysis unless stated otherwise. During the DeepLabCut tracking, seven animals were excluded due to technical difficulties (four NS and three CSDS were excluded). Statistical assumptions were then checked, in which the data were tested for normality using the Shapiro-Wilk test and QQ-plots and for heteroscedasticity using Levene's test. Data that violated these assumptions were analyzed using non-parametric tests. The time-course data was analyzed using the two-way ANOVA (parametric) or Kruskal-Wallis test (non-parametric) with time (days) as a within-subject factor and condition (NS vs. CSDS) as a between-subject factor, further posthoc analysis was performed using the Benjamini-Hochberg (BH) test (parametric) or the Wilcoxon test (non-parametric). P-values were adjusted for multiple testing using the Benjamini-Hochberg (BH) method. Three-group comparisons were analyzed using the one-way ANOVA (parametric) or Kruskal-Wallis test (non-parametric), and further posthoc analysis was performed using the BH test (parametric) or the Wilcoxon test (non-parametric). Two-group comparisons were analyzed using independent samples *t*-tests (parametric), Welch's tests (data are normalized but heteroscedastic), or Wilcoxon tests (non-parametric). Correlation analyses were performed using the Pearson correlation coefficient; outliers deviating more than 5 standard deviations from a fitted linear model were excluded from the analysis. The timeline and bar graphs are presented as mean \pm standard error of the mean. Data was considered significant at $p < 0.05$ (*), with $p < 0.01$ (**), $p < 0.001$ (***), $p < 0.0001$ (****).

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The authors declare that data supporting the findings of this study are available within the Article and Supplementary Information. Source data are provided with this paper.

Code availability

All data and the accompanying code to perform the analyses and creating the figures are available for download via the Max Planck DataShare services. The most recent version of DeepOF is hosted in a GitHub repository, and a Zenodo release of the version used in this manuscript (v0.4.6) is found under <https://doi.org/10.5281/zenodo.8013401>. The most recent stable version of DeepOF is available in PyPI. Full documentation and tutorials are available on read the docs.

References

- World Health Organisation. *Depression and Other Common Mental Disorders: Global Health Estimates*. Geneva: World Health Organization (2017).
- Cipriani, A. et al. Comparative efficacy and acceptability of 21 antidepressant drugs for the acute treatment of adults with major depressive disorder: a systematic review and network meta-analysis. *Focus* **16**, 420–429 (2018).
- American Psychiatric Association. *Diagnostic and Statistical Manual of Mental Disorders*. <https://doi.org/10.1176/appi.books.9780890425596> (American Psychiatric Association, 2013).
- Addabbo, T., Sarti, E. & Sciulli, D. Healthy life, social interaction and disability. *Qual. Quant.* **50**, 2609–2623 (2016).
- Giordano, G. N. & Lindstrom, M. The impact of changes in different aspects of social capital and material conditions on self-rated health over time: a longitudinal cohort study. *Soc. Sci. Med.* **70**, 700–710 (2010).
- Manchia, M. et al. The impact of the prolonged COVID-19 pandemic on stress resilience and mental health: a critical review across waves. *Eur. Neuropsychopharmacol.* **55**, 22–83 <https://doi.org/10.1016/j.euroneuro.2021.10.864> (2021).
- Santomauro, D. F. et al. Global prevalence and burden of depressive and anxiety disorders in 204 countries and territories in 2020 due to the COVID-19 pandemic. *Lancet* **398**, 1700–1712 [https://doi.org/10.1016/S0140-6736\(21\)02143-7](https://doi.org/10.1016/S0140-6736(21)02143-7) (2021).
- Gururajan, A., Reif, A., Cryan, J. F. & Slattery, D. A. The future of rodent models in depression research. *Nat. Rev. Neurosci.* **20**, 686–701 (2019).
- von Mücke-Heim, I.-A. et al. Introducing a depression-like syndrome for translational neuropsychiatry: a plea for taxonomical validity and improved comparability between humans and mice. *Mol. Psychiatry* **28**, 329–340 <https://doi.org/10.1038/s41380-022-01762-w> (2022).
- Golden, S. A., Covington, H. E., Berton, O. & Russo, S. J. A standardized protocol for repeated social defeat stress in mice. *Nat. Protoc.* **6**, 1183–1191 (2011).
- Russo, S. J. & Nestler, E. J. The brain reward circuitry in mood disorders. *Nat. Rev. Neurosci.* **14**, 609–625 (2013).
- Hollis, F. & Kabbaj, M. Social defeat as an animal model for depression. *ILAR J.* **55**, 221–232 (2014).
- Kudryavtseva, N. N., Bakshtanovskaya, I. V. & Koryakina, L. A. Social model of depression in mice of C57BL/6J strain. *Pharm. Biochem. Behav.* **38**, 315–320 (1991).
- Donahue, R. J., Muschamp, J. W., Russo, S. J., Nestler, E. J. & Carlezon, W. A. Effects of striatal δ fosb overexpression and ketamine on social defeat stress-induced anhedonia in mice. *Biol. Psychiatry* **76**, 550–558 (2014).
- Iñiguez, S. D. et al. Social defeat stress induces a depression-like phenotype in adolescent male c57BL/6 mice. *Stress* **17**, 247–255 (2014).
- Yoshida, K. et al. Chronic social defeat stress impairs goal-directed behavior through dysregulation of ventral hippocampal activity in male mice. *Neuropsychopharmacology* **46**, 1606–1616 (2021).
- Krishnan, V. et al. Molecular adaptations underlying susceptibility and resistance to social defeat in brain reward regions. *Cell* **131**, 391–404 (2007).
- Toth, I. & Neumann, I. D. Animal models of social avoidance and social fear. *Cell Tissue Res.* **354**, 107–118 (2013).
- Sturman, O. et al. Deep learning-based behavioral analysis reaches human accuracy and is capable of outperforming commercial solutions. *Neuropsychopharmacology* **45**, 1942–1952 (2020).
- Hånell, A. & Marklund, N. Structured evaluation of rodent behavioral tests used in drug discovery research. *Front. Behav. Neurosci.* **8**, 252 (2014).
- Goodwin, N. L., Nilsson, S. R. O. & Golden, S. A. Rage against the machine: advancing the study of aggression ethology via machine learning. *Psychopharmacology* **237**, 2569–2588 (2020).
- Pearson, B. L., Defensor, E. B., Blanchard, D. C. & Blanchard, R. J. C57BL/6J mice fail to exhibit preference for social novelty in the three-chamber apparatus. *Behav. Brain Res.* **213**, 189–194 (2010).
- Lorsch, Z. S. et al. Computational analysis of multidimensional behavioral alterations after chronic social defeat stress. *Biol. Psychiatry* **89**, 920–928 (2021).

24. Marks, M., et al Deep-learning based identification, pose estimation and end-to-end behavior classification for interacting primates and mice in complex environments. (2021).
25. Pereira, T. D. et al. Fast animal pose estimation using deep neural networks. *Nat. Methods* **16**, 117–125 (2019).
26. Hsu, A. I. & Yttri, E. A. B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. *Nat. Commun.* **12**, 5188 (2021).
27. Segalin, C. et al. The mouse action recognition system (MARS) software pipeline for automated analysis of social behaviors in mice. *Elife* **10**, e63720 (2021).
28. de Chaumont, F. et al. Real-time analysis of the behaviour of groups of mice via a depth-sensing camera and machine learning. *Nat. Biomed. Eng.* **3**, 930–942 (2019).
29. Mathis, A. et al. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nat. Neurosci.* **21**, 1281–1289 (2018).
30. Lauer, J. et al. Multi-animal pose estimation, identification and tracking with DeepLabCut. *Nat. Methods* **19**, 496–504 (2022).
31. Wiltischko, A. B. et al. Mapping sub-second structure in mouse behavior. *Neuron* **88**, 1121–1135 (2015).
32. Kabra, M., Robie, A. A., Rivera-Alba, M., Branson, S. & Branson, K. JAABA: interactive machine learning for automatic annotation of animal behavior. *Nat. Methods* **10**, 64–67 (2013).
33. Luxem, K. et al. Identifying behavioral structure from deep variational embeddings of animal motion. *Commun. Biol.* **5**, 1267 (2022).
34. Schmidt, M. V. & Koutsouleris, N. Promises and pitfalls of the new era of computational behavioral neuroscience. *Biol. Psychiatry* **89**, 845–846 (2021).
35. Miranda, L., Bordes, J., Gasperoni, S. & Lopez, J. P. Increasing resolution in stress neurobiology: from single cells to complex group behaviors. *Stress* **26**, 2186141 (2023).
36. Miranda, L., Bordes, J., Pütz, B., Schmidt, M. V. & Müller-Myhsok, B. DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data. *J. Open Source Softw.* **8**, 5394 (2023).
37. Nilsson, S. R. O. et al. Simple Behavioral Analysis (SimBA)—an open source toolkit for computer classification of complex social behaviors in experimental animals. *bioRxiv* 2020.04.19.049452 <https://doi.org/10.1101/2020.04.19.049452> (2020).
38. Lopez, J. P. et al. Single-cell molecular profiling of all three components of the HPA axis reveals adrenal ABCB1 as a regulator of stress adaptation. *Sci. Adv.* **7**, eabe4497 (2021).
39. Gheusi, G., Bluthé, R.-M., Goodall, G. & Dantzer, R. Social and individual recognition in rodents: methodological aspects and neurobiological bases. *Behav. Process.* **33**, 59–87 (1994).
40. Brudzynski, S. M. Ethotransmission: communication of emotional states through ultrasonic vocalization in rats. *Curr. Opin. Neurobiol.* **23**, 310–317 (2013).
41. Hamasato, E. K., Lovelock, D., Palermo-Neto, J. & Deak, T. Assessment of social behavior directed toward sick partners and its relation to central cytokine expression in rats. *Physiol. Behav.* **182**, 128–136 (2017).
42. Rogers-Carter, M. M., Djerdjaj, A., Culp, A. R., Elbaz, J. A. & Christianson, J. P. Familiarity modulates social approach toward stressed conspecifics in female rats. *PLoS One* **13**, e0200971 (2018).
43. Morel, C. et al. Midbrain projection to the basolateral amygdala encodes anxiety-like but not depression-like behaviors. *Nat. Commun.* **13**, 1532 (2022).
44. Ye, S., Mathis, A. & Mathis, M. W. Panoptic animal pose estimators are zero-shot performers. (2022).
45. Schneider, S., Lee, J. H. & Mathis, M. W. Learnable latent embeddings for joint behavioral and neural analysis. *Nature* **617**, 360–368 (2022).
46. Sun, J. J. et al. The multi-agent behavior dataset: mouse dyadic social interactions. (2021).
47. Savitzky, A. & Golay, M. J. E. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.* **36**, 1627–1639 (1964).
48. Buitinck, L. et al. API design for machine learning software: experiences from the scikit-learn project. (2013).
49. Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002).
50. Lemaître, G., Nogueira, F. & Aridas, C. K. Imbalanced-learn: a Python toolbox to tackle the curse of imbalanced datasets in machine learning. *jmlr. org.* **18**, 1–5 (2017).
51. Kleinberg, J. Bursty and hierarchical structure in streams. In: *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 91–101. <https://doi.org/10.1145/775060.775061> (2002).
52. Zhou, J. et al. Graph neural networks: a review of methods and applications. *AI Open* **1**, 57–81 (2020).
53. Jiang, Z., Zheng, Y., Tan, H., Tang, B. & Zhou, H. Variational deep embedding: an unsupervised and generative approach to clustering. In: *Proc. Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)* (2016).
54. Manduchi, L. et al. A Deep variational approach to clustering survival data. (2021).
55. Lafabregue, B., Weber, J., Gançarski, P. & Forestier, G. End-to-end deep representation learning for time series clustering: a comparative study. *Data Min. Knowl. Discov.* **36**, 29–81 (2022).
56. Yu, B., Yin, H. & Zhu, Z. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: *Proc. Twenty-Seventh International Joint Conference on Artificial Intelligence.* 3634–3640. <https://doi.org/10.24963/ijcai.2018/505> (2018).
57. Jiang, X., Zhu, R., Ji, P. & Li, S. Co-Embedding of Nodes and Edges With Graph Neural Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **45**, 7075–7086 (2023).
58. Dozat, T. Incorporating Nesterov Momentum into Adam. Preprint at (2016).
59. Goodwin, N. L., Nilsson, S. R. O., Choong, J. J. & Golden, S. A. Toward the explainability, transparency, and universality of machine learning for behavioral classification in neuroscience. *Curr. Opin. Neurobiol.* **73**, 102544 (2022).
60. Lundberg, S. & Lee, S.-I. A Unified Approach to Interpreting Model Predictions. (2017).
61. Dorogush, A. V., Ershov, V. & Gulin, A. CatBoost: gradient boosting with categorical features support. (2018).
62. Wagner, K. V. et al. Differences in FKBP51 regulation following chronic social defeat stress correlate with individual stress sensitivity: influence of paroxetine treatment. *Neuropsychopharmacology* **37**, 2797–2808 (2012).
63. Berton, O. et al. Essential role of BDNF in the mesolimbic dopamine pathway in social defeat stress. *Science (1979)* **311**, 864–868 (2006).
64. Tsankova, N. M. et al. Sustained hippocampal chromatin regulation in a mouse model of depression and antidepressant action. *Nat. Neurosci.* **9**, 519–525 (2006).
65. Guilloux, J.-P., Seney, M., Edgar, N. & Sibille, E. Integrated behavioral z-scoring increases the sensitivity and reliability of behavioral phenotyping in mice: Relevance to emotionality and sex. *J. Neurosci. Methods* **197**, 21–31 (2011).
66. R: The R Project for Statistical Computing. <https://www.r-project.org/>.
67. Zhao, Y. et al. Social rank-dependent effects of testosterone on huddling strategies in mice. *iScience* **26**, 106516 (2023).

Acknowledgements

The authors thank the DeepLabCut development team for creating and maintaining the DeepLabCut software. In addition, the authors thank Margherita Springer for the language editing of the manuscript and Max Pöhlmann for the design of the mouse illustrations in Fig. 2a. This study is supported by the Kids2Health grant of the Federal Ministry of Education and Research (01GL1743C, M.V.S.), and the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement (813533; L.M.).

Author contributions

JB and MVS conceived the study. L.M. wrote the DeepOF module, with primary technical assessment from F.A., B.P., and B.M.M. J.B. and M.R. performed the experiments. L.M.B., L.v.D., C.E., L.D., and S.M. assisted with the experiments. J.B. and L.M. analyzed the data and wrote the first version of the manuscript. B.P. worked on figure design. J.B. created the mouse illustrations in Fig. 1. S.N., J.H., E.L.N., and K.J.R. assisted with manual behavioral data tracking and analysis for data benchmarking purposes. All authors contributed to the revision of the manuscript.

Funding

Open Access funding enabled and organized by Projekt DEAL.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41467-023-40040-3>.

Correspondence and requests for materials should be addressed to Bertram Müller-Myhsok or Mathias V. Schmidt.

Peer review information *Nature Communications* thanks David Slattery, Johannes Bohacek and the other, anonymous, reviewer for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at <http://www.nature.com/reprints>

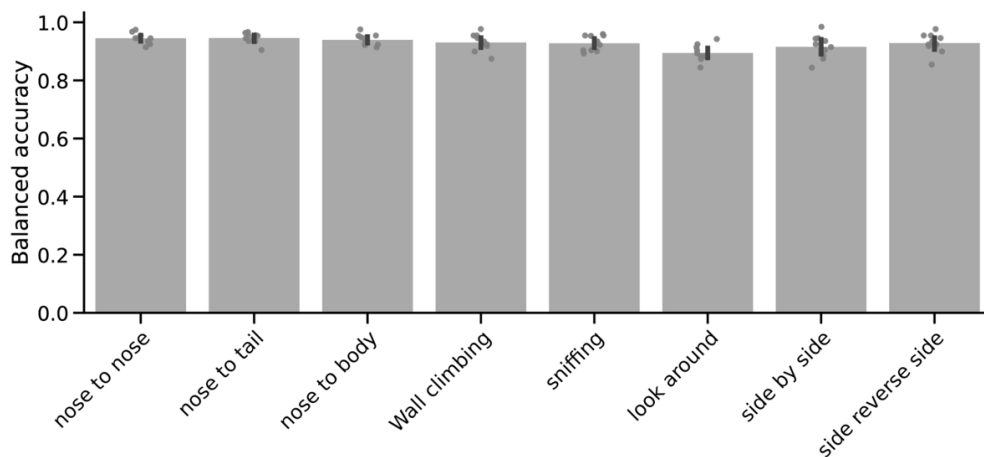
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

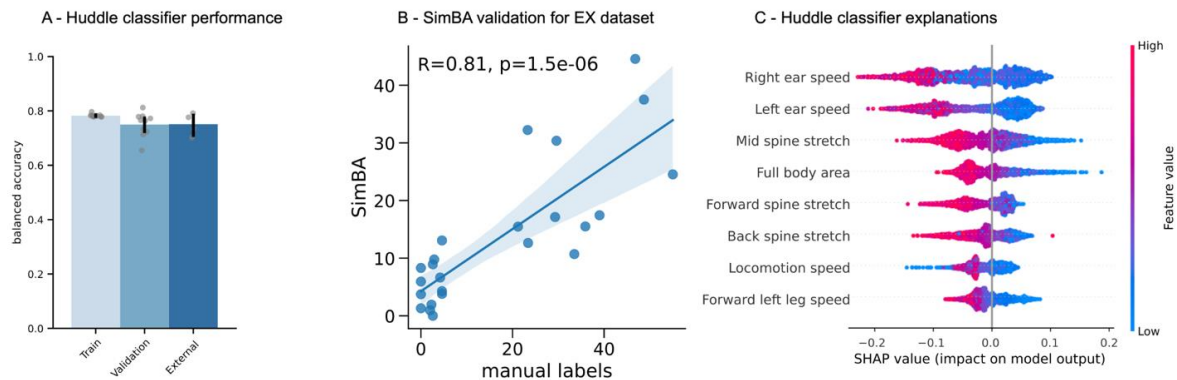
© The Author(s) 2023

Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress

Supplemental material



Supplemental figure 1. Validation of rule-based annotated behaviors. 10 out of 53 videos were manually labeled for all annotators (excluding stopped-and-huddled, see supplemental figure 2) using the Colabeler software (v2.0.4). Balanced accuracy between manual labels and predicted binary outcomes (presence or absence of a given trait at a given time) is reported. Bars represent the mean \pm standard deviation across all 10 videos (N=10). Source data are provided as a Source Data file.



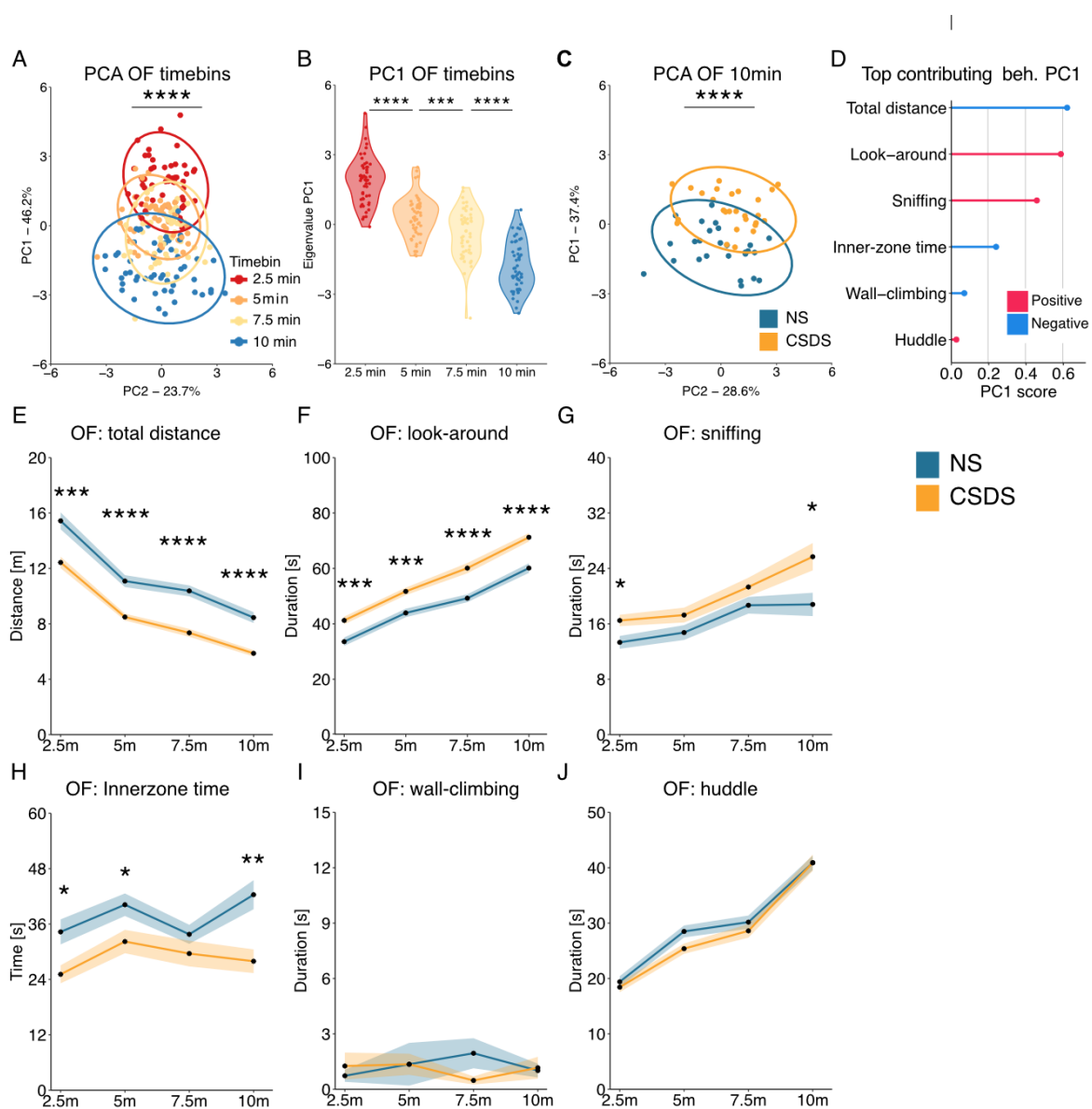
Supplemental figure 2. Validation of stopped-and-huddled classifier. A) Bar charts (mean \pm standard deviation) showing balanced accuracy performance for the huddle classifier provided with the supervised pipeline within DeepOF. A total of 567367 video frames were either manually labeled (for the SI, OF, and SA datasets) or pseudo-labeled using SimBA (EX dataset) for the stopped-and-huddled trait using the labeling tool provided with SimBA v1.31.1. Labelling was conducted in four independent datasets (SI, OF, SA, and EX; see the animals' section in materials and methods for details), and two validation tasks were conducted, marked as "Validation" and "External" respectively. First, a 10-fold stratified cross-validation loop was executed within the SI dataset (which has the most labels, see supplemental table 2 for details), to test for overfitting and generalization within a single dataset. Balanced accuracy results were 0.78 ± 0.005 and 0.75 ± 0.046 for the training and validation sets, respectively (N=10). Second, a leave-one-dataset out cross-validation was conducted across all four datasets, to test whether the model can generalize to novel settings (different bedding, different arenas, different labs). A balanced accuracy of 0.75 ± 0.04 was reported (N=4). B) SimBA validation of the classifiers used for pseudo-labelling in the external dataset. Correlation between total behavior duration (in seconds) in manual and predicted labels shown for all 24 videos (N=24). Both sets show a Pearson correlation coefficient $\rho=0.81$, which significantly deviates from zero (p -value= $1.5e-6$). Error bands represent the 95% confidence interval. C) SHAP analysis of the deployed model (trained in the whole dataset, with all concatenated four sites). The top 8 features are displayed of a total of 26 features including distances between body parts, speeds, and areas. Results show low head movement, low spine stretch, low body area, and low locomotion speed as the most important features for the model, which goes in line with the accepted definition of the behavior. Source data are provided as a Source Data file.

Supplemental table 1. Default thresholds used by the annotation pipeline in DeepOF

Annotated trait	Rule	Default threshold in DeepOF
Nose-to-nose	Nose to nose distance	< 25 mm
Nose-to-tail	Nose to tail distance	< 25 mm
Nose-to-body	Nose to any other body part	< 25 mm
Side-by-side	Nose to nose distance	< 45 mm
	Tail to tail distance	< 45 mm
Side-reverse-side	Nose to tail distance	< 45 mm
Wall-climbing	Nose reach beyond walls	> 10 mm
Sniffing	Nose distance to object	< 10 mm
	Nose speed	> 50 mm/s
	Locomotion speed	< 50 mm/s
Look-around	Locomotion speed	< 50 mm/s
	Nose speed	> 50 mm/s

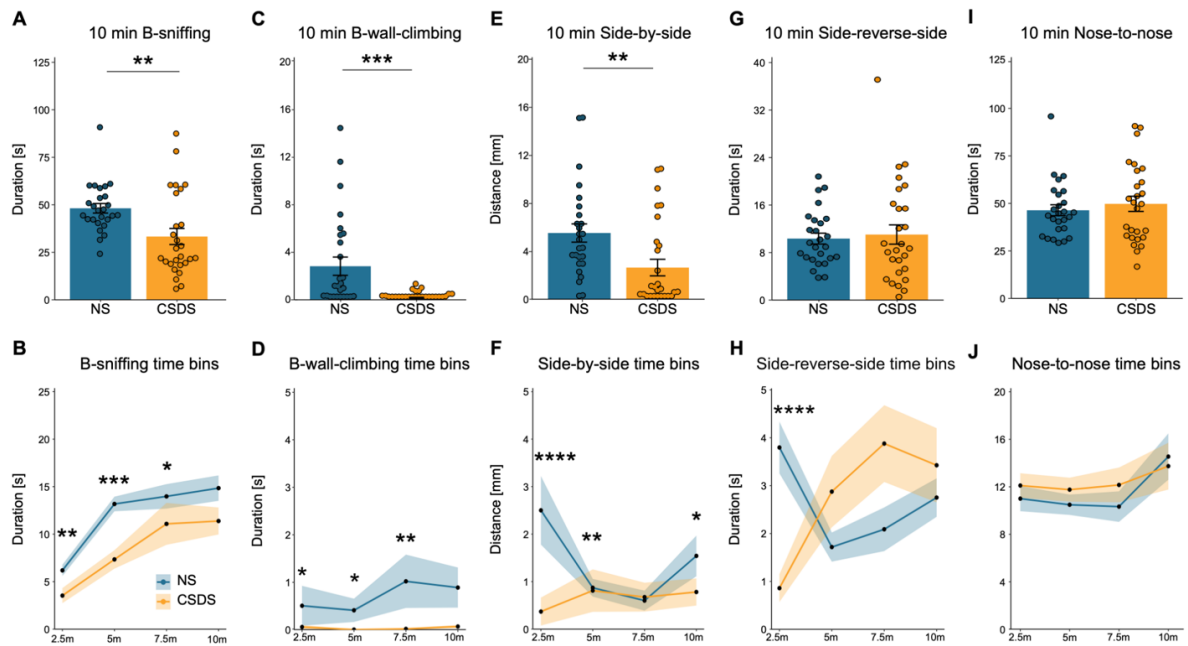
Supplemental table 2. Datasets used in the current study

Dataset name	Experiment code	Number of videos	Frame rate	Video length	Labeled frames (stopped-and-huddled)	Prevalence (stopped-and-huddled)
Social interaction (SI)	1	53	25	10 min 15000 frames	299.350	10.83%
Open field (OF)	1	53	25	10 min 15000 frames	179.979	2.75%
Social avoidance (SA)	1	120	13	2.5 min 1950 frames	22.488	4.36%
Social interaction for SA resiliency (figure S6)	2	64	30	10 min 18000 frames	0	-
Social interaction (external)	3	20	30	1.5 min 2730 frames	65.550	14.95%
Total	-	310	-	-	567.367	8.49%

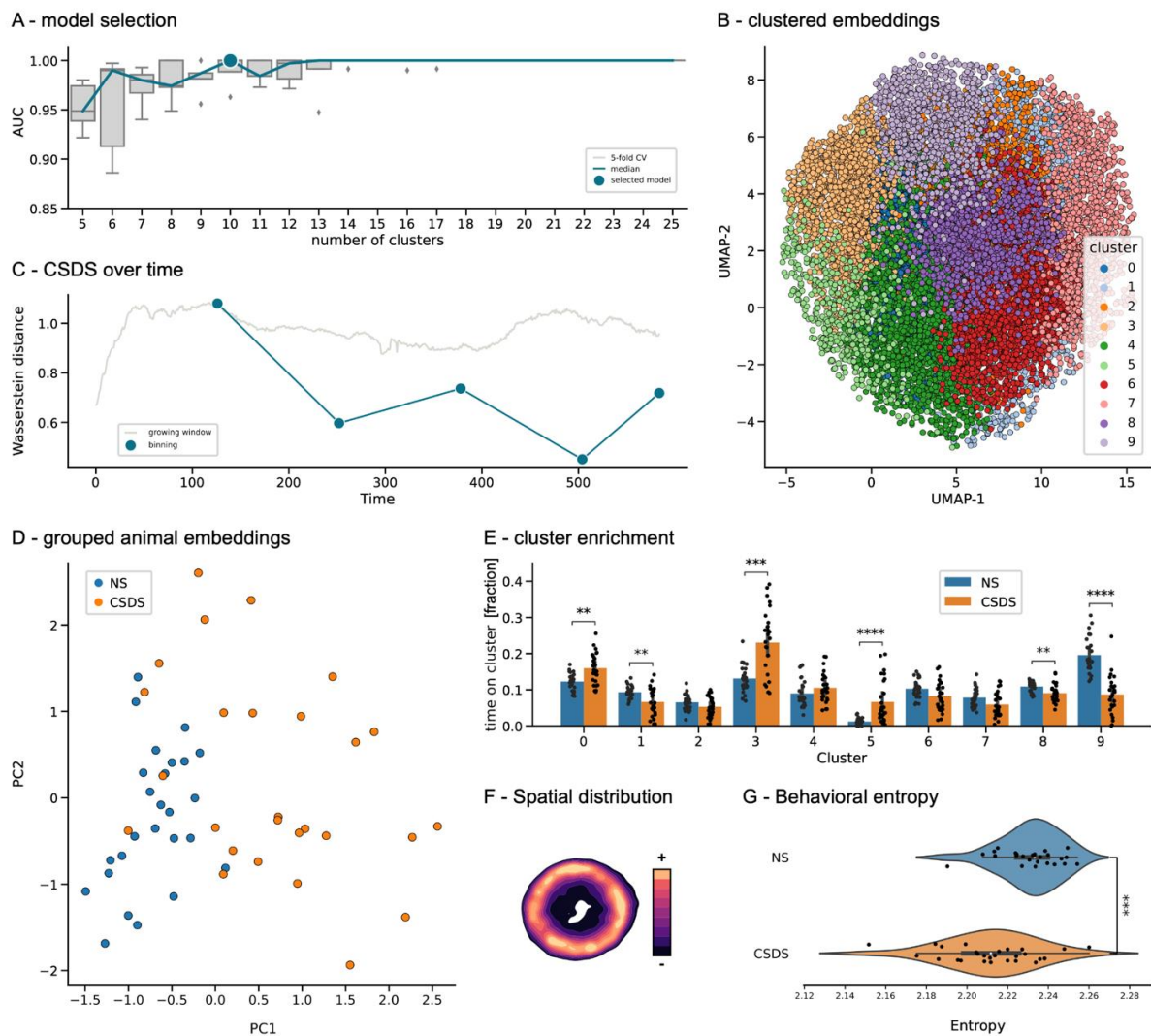


Supplemental figure 3. DeepOF behavioral classifiers in the open field task. A) The OF PCA time bins show a significant main effect (one-way ANOVA: $F(3,208)=129.12$, $p=2.97e-47$). B) Benjamini-hochberg (BH) posthoc shows that the time bins are significantly different from each other (2.5vs5, $p=3.93e-14$; 5vs7.5, $p=0.0003$, 7.5vs10, $p=3.1e-12$). C) The 10min OF PCA analysis shows a significant difference between conditions; independent samples t -test: $T(51)=-7.23$, $p=2.37e-9$. Data consisted of all the individual DeepOF behavioral classifiers, as listed in Figure 1C. D) The ranked behaviors on the PC1 using the corresponding rotated loading scores. E) The total distance was lower in CSDS animals; posthoc BH: 2.5 min $T(51)=16.89$, $p=0.0001$, 5 min $T(51)=28.28$, $p=3.13e-6$, 7.5 min $T(51)=39.59$, $p=2.86e-7$, and 10 min $T(51)=33.77$, $p=8.1e-7$. Two-way ANOVA on condition: $F(1,208)=92.586$, $p=2.31e-18$, time: $F(1,208)=265.77$, $p=4.85e-39$, condition \times time: $F(1,208)=0.10$, $p=0.75$). F) Look-around was higher in CSDS

animals; posthoc BH: 2.5 min ($T(51)=14.08$, $p=0.0004$), 5 min ($T(51)=14.84$, $p=0.0004$), 7.5 min ($T(51)=21.65$, $p=4.7e-5$), and 10 min ($T(51)=23.25$, $p=4.7e-5$). Two-way ANOVA on condition: $F(1,208)=74.04$, $p=1.9e-15$, time: $F(1,208)=356.65$, $p=5.4e-47$, condition \times time: $F(1,208)=1.90$, $p=0.17$). G) Sniffing was higher in CSDS animals for the 2.5- and 10 min time bins; posthoc Wilcoxon: $W=199.5$, $p=0.023$; $W=210$, $p=0.023$, respectively. The 5- and 7.5 min were not altered ($W=258$, $p=0.13$, and $W=307$, $p=0.44$, respectively). Kruskal-Wallis test 2.5 min: $H(1)=7.27$, $p=0.024$, 5 min: $H(1)=2.74$, $p=0.13$, 7.5 min: $H(1)=0.6$, $p=0.43$, and 10 min: $H(1)=6.29$, $p=0.024$. H) The inner zone time was lowered in CSDS animals for the 2.5, 5, and 10 min time bins; posthoc BH: $T(51)=7.70$, $p=0.016$, $T(51)=5.16$, $p=0.036$, $T(51)=12.74$, $p=0.0032$, respectively). The 7.5 min was not altered ($p=0.24$). Two-way ANOVA on condition: $F(1,208)=24.04$, $p=1.9e-6$, time: $F(1,208)=2.07$, $p=0.15$, condition \times time: $F(1,208)=0.53$, $p=0.47$). I) Climbing did not reveal any difference using the Kruskal-Wallis test. J) Huddle did not reveal any difference using the Kruskal-Wallis test. The PCA graphs are provided with a 95% confidence ellipse and all individual samples as points. Further PC1 analyses are represented with a violin plot and all individual samples as points. The timeline graphs are presented as mean \pm standard error of the mean. $N=26$ for NS and $n=27$ for CSDS in panels A-J. Source data are provided as a Source Data file.

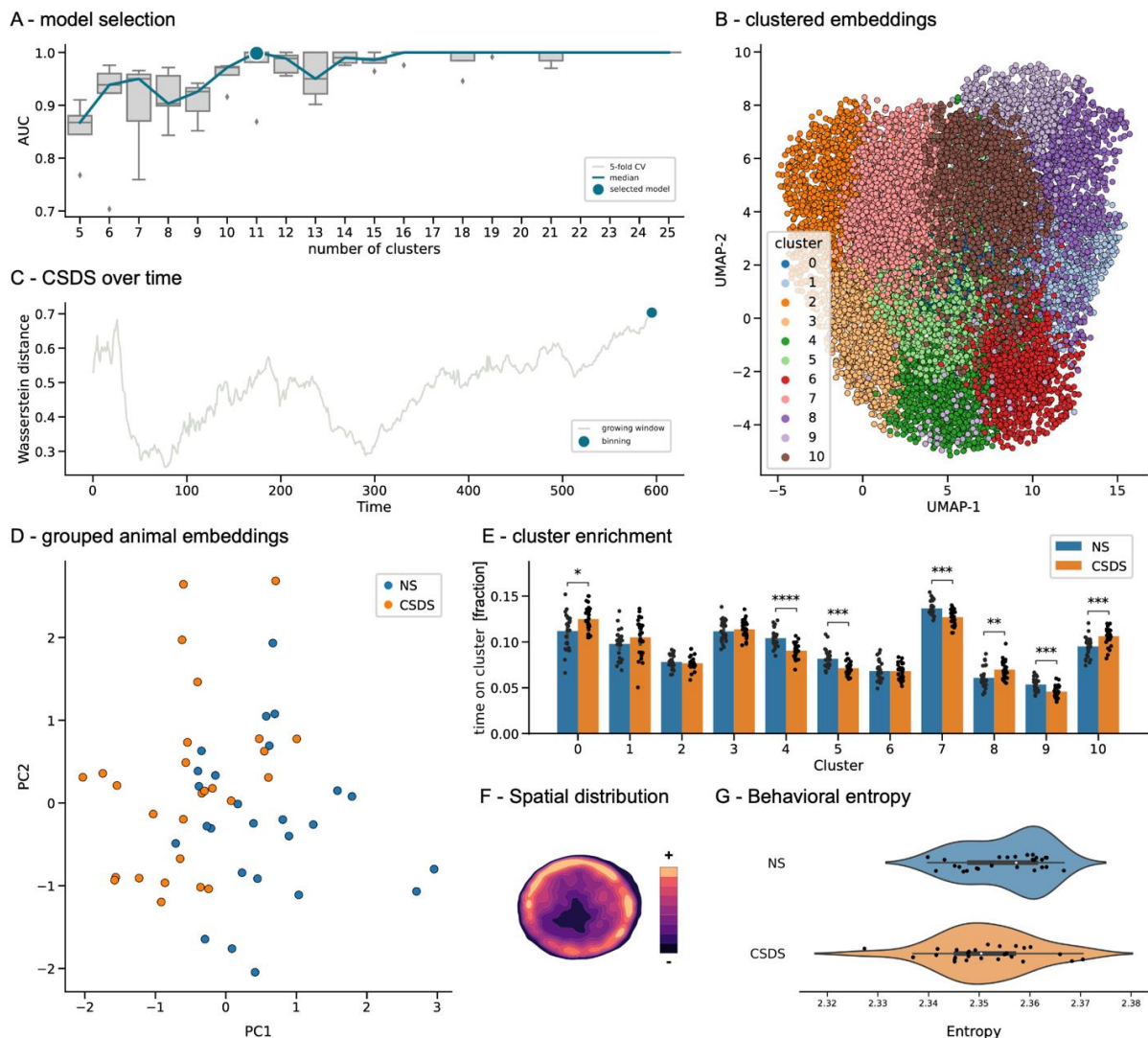


Supplemental Figure 4. DeepOF other behavioral classifiers in the social interaction task for 10 min duration. A) B-sniffing is lower in CSDS animals. Independent samples *t*-test: $T(51)=2.99$, $p=0.004$. B) Wilcoxon posthoc analysis revealed that B-sniffing was lower in CSDS animals for the 2.5 min ($W=538$, $p=0.002$), 5 min ($W=576$, $p=0.0003$), and 7.5 min ($W=499$, $p=0.012$), but not the 10 min ($W=456$, $p=0.06$). Kruskal-Wallis test: 2.5 min: $p=0.002$, 5 min: $p=0.0003$, 7.5 min: $p=0.012$, and 10 min: $p=0.06$. C) B-wall-climbing is lower in stressed animals. Wilcoxon test: $W=540$, $p=0.0004$. D) Wilcoxon posthoc analysis revealed that B-wall-climbing was lower in stressed animals for the 2.5 min ($W=441$, $p=0.03$), the 5 min ($W=435$, $p=0.03$), and the 7.5 min ($W=506$, $p=0.002$), but not the 10 min ($W=393$, $p=0.37$). Kruskal-Wallis test: 2.5 min: $p=0.03$, 5 min: $p=0.03$, 7.5 min: $p=0.002$, and 10 min: $p=0.37$. E) Side-by-side is lower in CSDS animals. Wilcoxon test: $W=522.5$, $p=0.0023$. F) Wilcoxon posthoc analysis revealed that Side-by-side was lower in CSDS animals for the 2.5 min ($W=581$, $p=5.48e-5$), the 5 min ($W=521.5$, $p=0.003$), and the 10 min ($W=491.5$, $p=0.02$), but not the 7.5 min ($W=405$, $p=0.32$). Kruskal-Wallis test: 2.5 min: $p=5.28e-5$, 5 min: $p=0.003$, 7.5 min: $p=0.32$, and 10 min: $p=0.02$. G) Side-reverse-side is not altered between conditions. Wilcoxon test: $W=365$, $p=0.81$. H) Wilcoxon posthoc analysis revealed that Side-reverse-side was lower in CSDS animals for the 2.5 min time bin ($W=628$, $p=3.36e-6$), but not the 5-, 7.5-, and 10 min time bins ($W=337.5$, $p=1$; $W=292.5$, $p=0.60$; $W=351$, $p=1$, respectively). Kruskal-Wallis test: 2.5 min: $p=3.21e-6$, 5 min: $p=1$, 7.5 min: $p=0.60$, and 10 min: $p=1$. I) Nose-to-nose is not altered between conditions. Wilcoxon test: $W=326$, $p=0.67$. J) No further significant differences were observed in the Nose-to-nose time bins. The timeline and bar graphs are presented as mean \pm standard error of the mean and all individual samples as points. $N=26$ for NS and $N=27$ for CSDS in panels A-J. Source data are provided as a Source Data file.



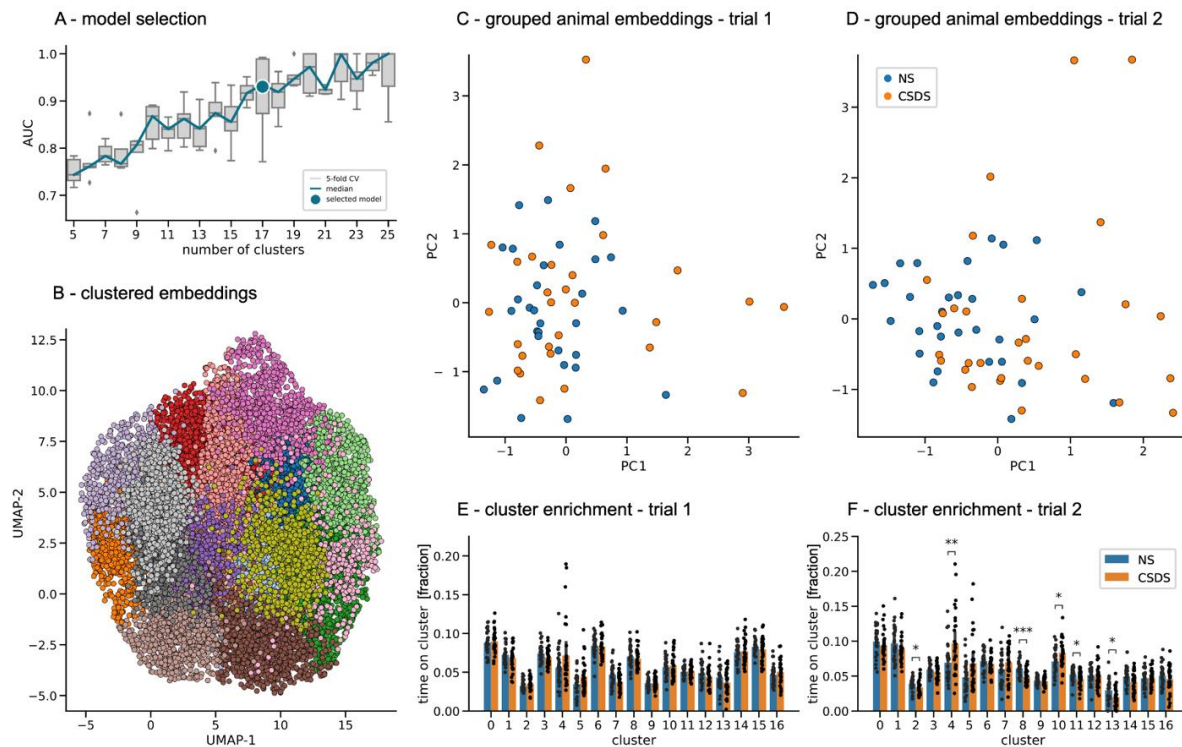
Supplemental Figure 5. Multi-animal unsupervised analyses identify different two-mice behavioral patterns between arenas containing stressed and non-stressed mice during the SI task. A) Cluster selection pipeline results, reporting the area under the ROC curve from a logistic regression classifier discriminating between conditions. A 10-component solution (from a range between 5 and 25) was selected as optimal in a 5-fold (N=5) cross-validation loop (see methods for details). B) Embeddings by time point obtained using DeepOF's unsupervised pipeline. Different colors correspond to different clusters. Dimensionality was further reduced from the original 8-dimensional embeddings using UMAP for visualization purposes. C) Optimal binning of the videos was obtained as the Wasserstein distance between the global animal embeddings of both conditions across a growing window, between the first 10 to 600 seconds for each video at one-second intervals (grey curve). Higher values correspond to larger behavioral differences across conditions. A maximum was observed at 124 seconds, close to the 126 seconds obtained with the single-animal embeddings, and to the stipulated 150 seconds selected based on the SA task literature. The dark green curve depicts the Wasserstein distance across all subsequent non-overlapping bins with optimal length. The decay observed across time is consistent with the hypothesized arousal period in the CSDS cohort, which can be detected also embedding the two-mice system as a whole. D) Representation of the global animal embeddings for the optimally discriminant bin

(124 seconds) per experimental video colored by condition (see methods for details). E) Cluster enrichment per experimental condition (N=26 for NS and N=27 for CSDS) in the first optimal bin (first 124 seconds). Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using the Benjamini-Hochberg method across both clusters and bins (significant differences observed in clusters 0: $U=1.7e+2$, $p=1.2e-3$, 1: $U=4.9e+2$, $p=8.5e-3$, 3: $U=1.4e+2$, $p=1.4e-4$, 5: $U=8.4e+1$, $p=2.1e-6$, 8: $U=5.3e+2$, $p=1.2e-3$, 9: $U=6.7e+2$, $p=1.4e-8$). Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. F) Example heatmap depicting spatial distribution across all experiments (in both conditions) for all clusters. Specific heatmaps for all individual clusters are available in supplemental figure 13). G) Behavioral entropy scores per condition. NS animals show a significantly higher entropy than CSDS animals, which can be attributed to a less predictable exploration of the behavioral space ($U=5.44e+2$, $p=6.15e-4$, N=26 for NS and N=27 for CSDS). Moreover, and in accordance with these results, behavioral entropy shows a significant negative correlation with the presented stress physiology Z-score (supplemental figure 15B). Source data are provided as a Source Data file. Box plots in panels A and G show the median and the inter-quartile range. Whiskers show the full range, excluding outliers as a function of the inter-quartile range.

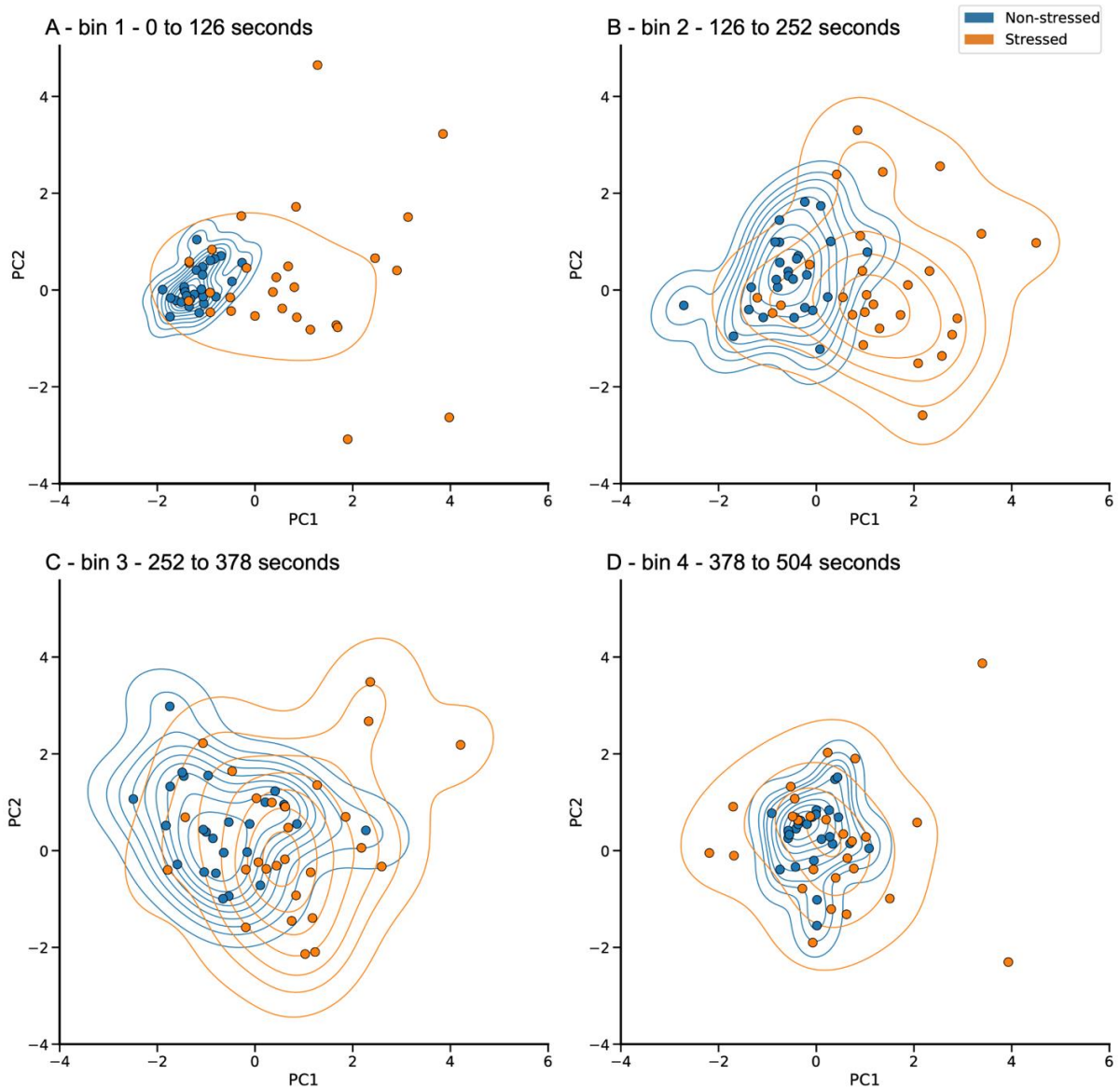


Supplemental Figure 6. Single-animal unsupervised analyses identify different behavioral patterns between stressed and non-stressed mice during the OF task. A) Cluster selection pipeline results, reporting the area under the ROC curve from a logistic regression classifier discriminating between conditions. An 11-component solution (from a range between 5 and 25) was selected as optimal in a 5-fold (N=5) cross-validation loop (see methods for details). B) Embeddings by time point obtained using DeepOF's unsupervised pipeline. Different colors correspond to different clusters. Dimensionality was further reduced from the original 8-dimensional embeddings using UMAP for visualization purposes. C) Optimal binning of the videos was obtained as the Wasserstein distance between the global animal embeddings of both conditions across a growing window, between the first 10 to 600 seconds for each video at one-second intervals (grey curve). Higher values correspond to larger behavioral differences across conditions. A maximum was observed at 595 seconds (green dot), which is consistent with the hypothesized lack of an arousal period in the CSDS cohort in an open field setting with no conspecific. D) Representation of the global animal embeddings for the optimally discriminant bin (595 seconds) per experimental video colored by condition (see methods for details). E) Cluster enrichment per

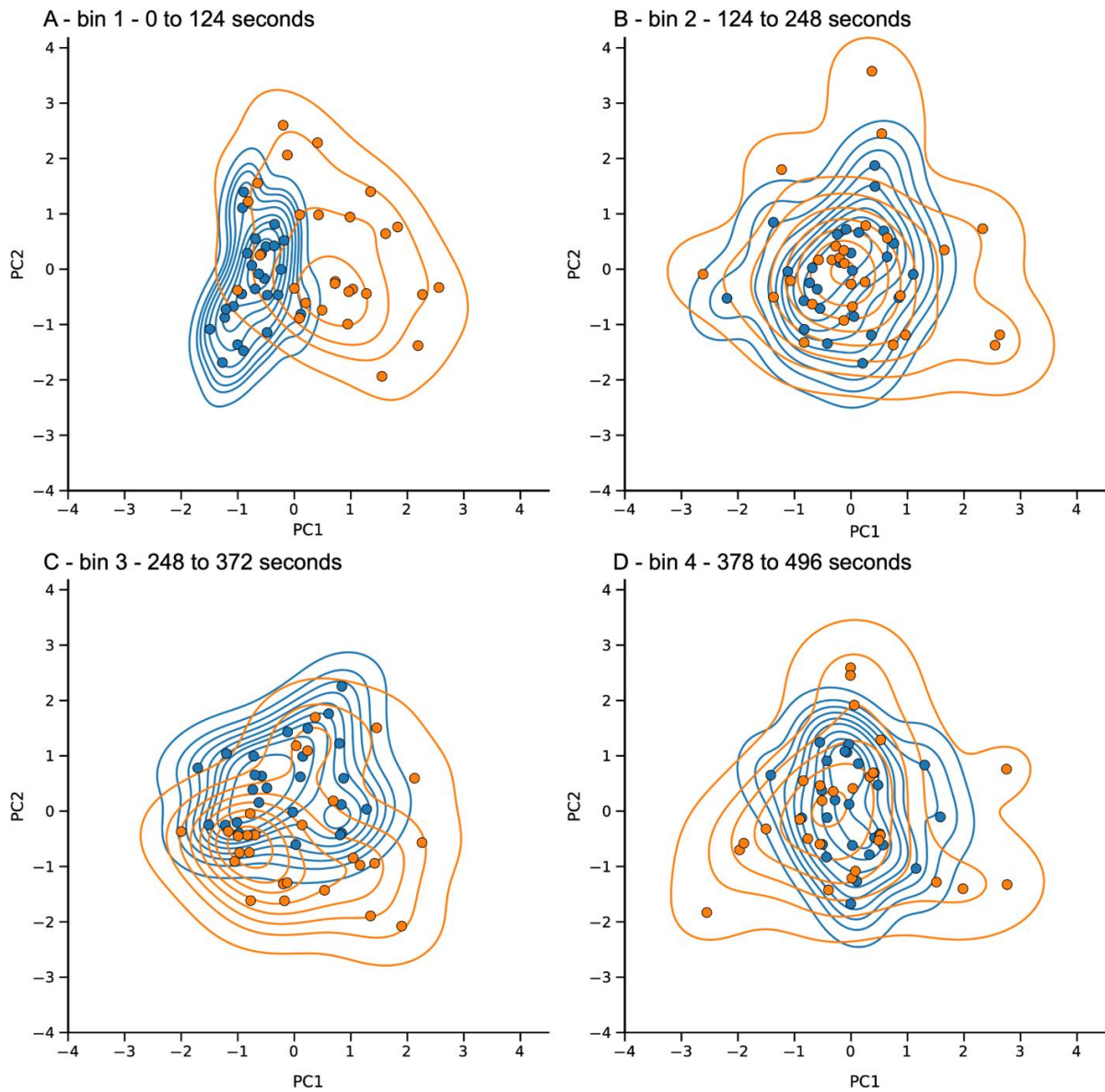
experimental condition (N=26 for NS and N=27 for CSDS) in the first optimal bin (first 595 seconds). Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using the Benjamini-Hochberg method across both clusters and bins (significant differences observed in clusters 0: $U=2.2e+2$, $p=2.02e-2$, 4: $U=6.1e+2$, $p=5.7e-6$, 5: $U=5.7e+2$, $p=1.3e-4$, 7: $U=5.4e+1$, $p=9.9e-4$, 8: $U=1.8e+2$, $p=2.3e-3$, 9: $U=5.5e+2$, $p=3.7e-4$, and 10: $U=1.5e+2$, $p=2.6e-4$). Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. F) Example heatmap depicting spatial distribution across all experiments (in both conditions) for all clusters. Specific heatmaps for all individual clusters are available in supplemental figure 14). G) Behavioral entropy scores per condition. No significant differences are detected between conditions ($U=4.44e+2$, $p=9.98e-2$, N=26 for NS and N=27 for CSDS). Moreover, and in accordance with these results, no significant correlation with the presented stress physiology Z-score was found (supplemental figure 15C). Source data are provided as a Source Data file. Box plots in panels A and G show the median and the inter-quartile range. Whiskers show the full range, excluding outliers as a function of the inter-quartile range.



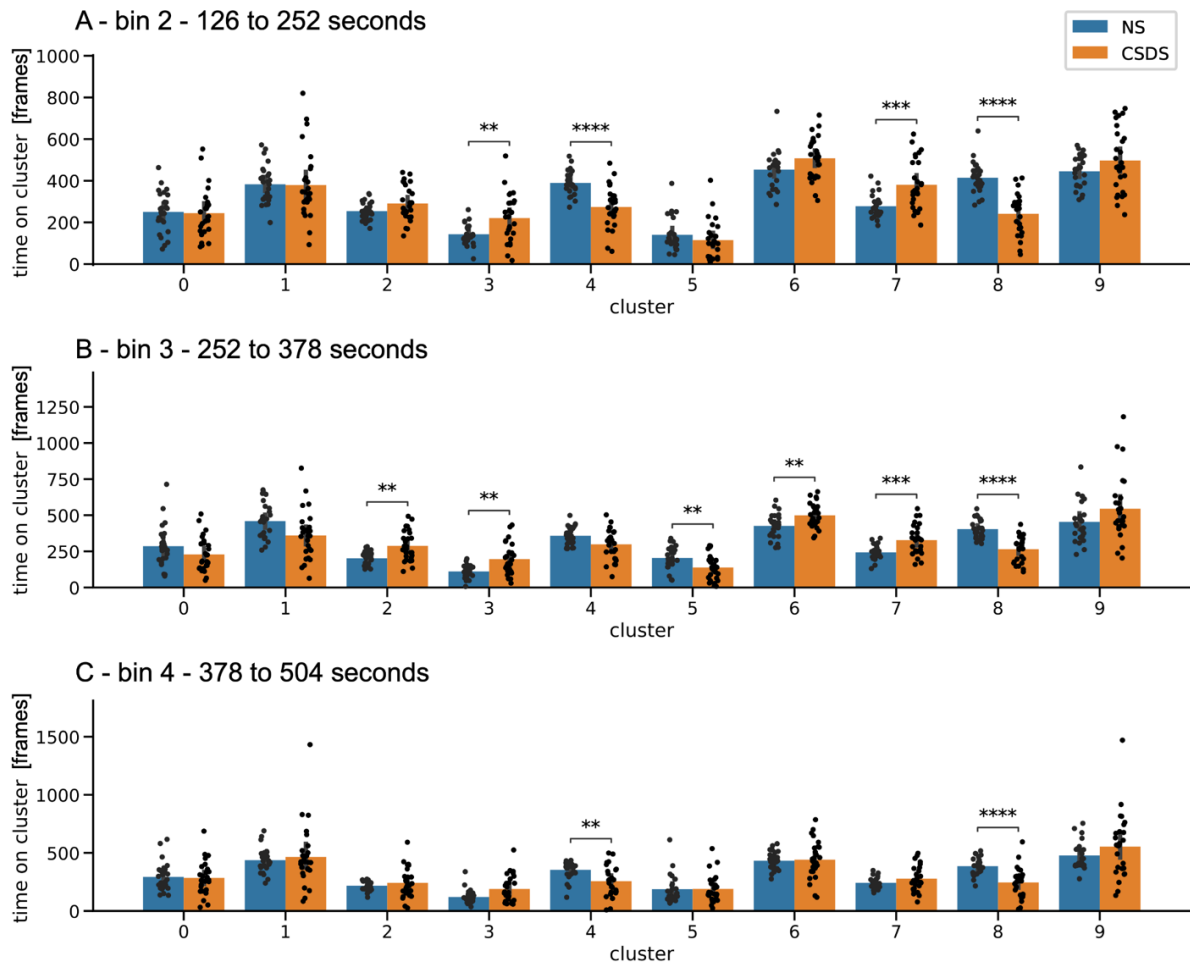
Supplemental Figure 7. Single-animal unsupervised analyses identify mild behavioral differences between stressed and non-stressed mice during the SA task. A) Cluster selection pipeline results. Models ranging from 5 to 25 clusters were trained in a 5-fold ($N=5$) cross-validation loop using data from both trials together. Area under the ROC curve from a logistic regression classifier discriminating between conditions on the global animal embeddings representing the differential population of each cluster across trials is reported. A 17-component solution was selected as the smallest whose median performance deviated less than one standard deviation from the maximum reached median across all clusters. Boxes in the box plots show the median performance and the inter-quartile range of the data. Whiskers show the full range of the data, excluding outliers as a function of the inter-quartile range. B) Embeddings by time point obtained using DeepOF's unsupervised pipeline. Different colors correspond to different clusters. Dimensionality was further reduced from the original 8-dimensional embeddings using UMAP for visualization purposes. C-D) Representation of the global animal embeddings per experimental video colored by condition, for SA trials one (without conspecific in the cage) and two (with conspecific in the cage). In panel C, as expected, the distributions are further apart. E-F) Cluster enrichment per experimental condition for both SA trials ($N=30$ for NS and $N=30$ for CSDS). As expected, trial one shows no significant differences, whereas trial two yields six significantly differentially expressed clusters. Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using the Benjamini-Hochberg method across both clusters (significant differences for trial two observed in clusters 2: $U=6.1e+2$, $p=1.4e-2$, 4: $U=2.6e+2$, $p=7.3e-6$, 8: $U=7.01e+2$, $p=2.1e-4$, 10: $U=2.8e+2$, $p=1.4e-2$, 11: $U=6.1e+2$, $p=1.7e-2$, and 13: $U=6.1e+2$, $p=1.8e-2$. Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. Source data are provided as a Source Data file.



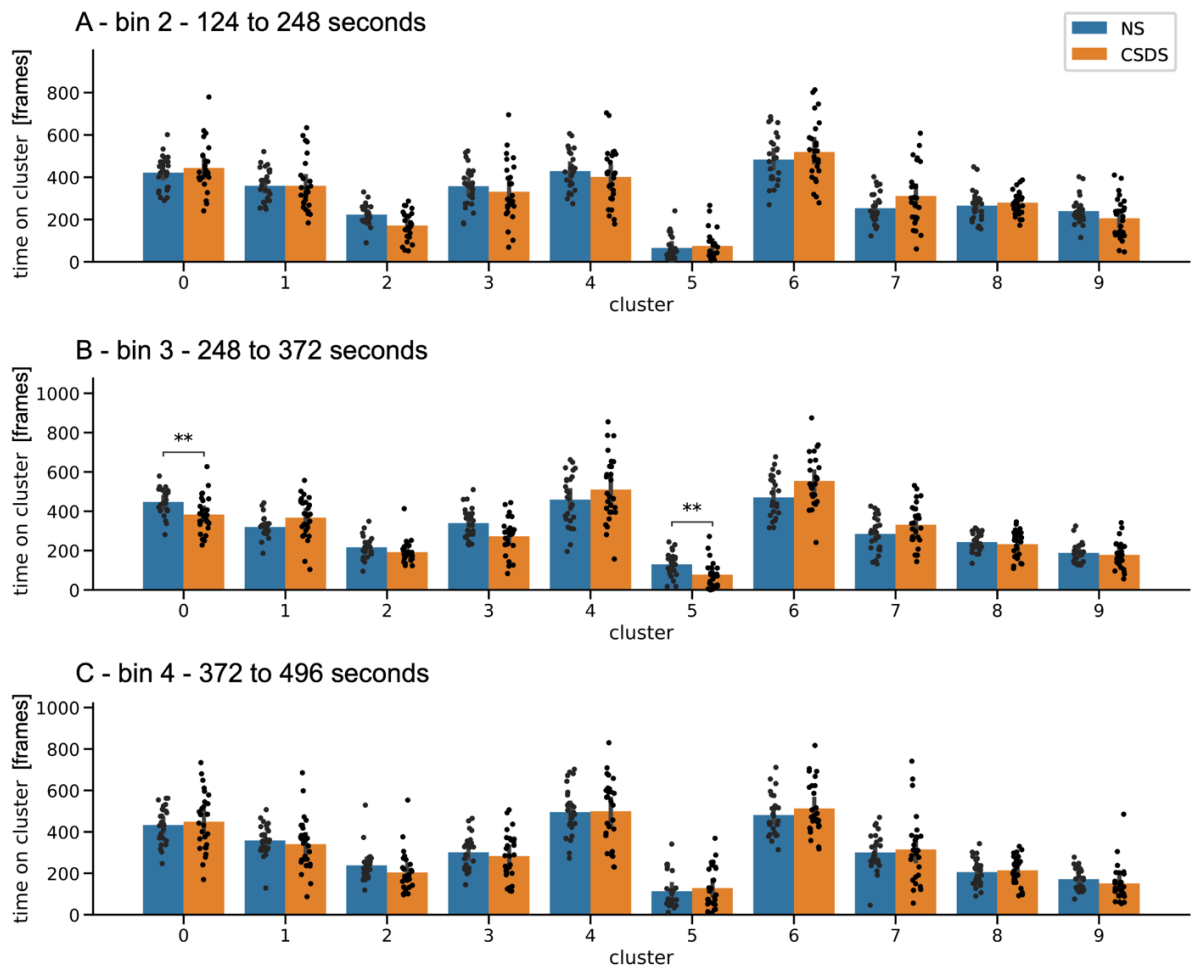
Supplemental figure 8. Global single-animal embeddings across non-overlapping time bins in the SI dataset. A-D) 10-dimensional global single-animal embeddings were obtained as the time proportion spent on each of the 10 clusters in the selected model for the single-animal SI task. Panels A to D show how the distributions matching NS and CSDS animals get closer and closer across non-overlapping consecutive time bins (as quantified using Wasserstein distance in the first four points shown in dark green in figure 6B). The last bin was excluded for visualization purposes. Source data are provided as a Source Data file.



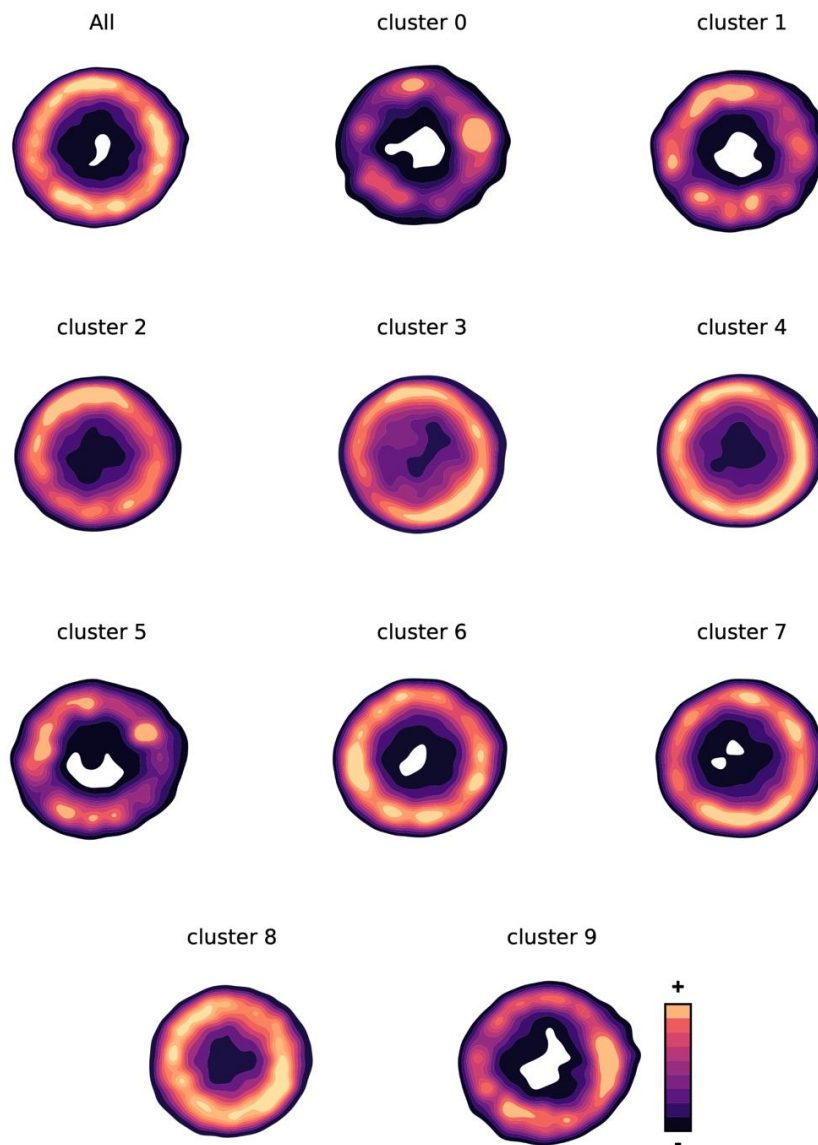
Supplemental figure 9. Global multi-animal embeddings across non-overlapping time bins in the SI dataset. A-D) 10-dimensional global single-animal embeddings were obtained as the time proportion spent on each of the 10 clusters in the selected model for the multi-animal SI task. Panels A to D show how the distributions matching NS and CSDS animals get closer across non-overlapping consecutive time bins (as quantified using Wasserstein distance in the first four points shown in dark green in supplemental figure 9B). The last bin was excluded for visualization purposes. Source data are provided as a Source Data file.



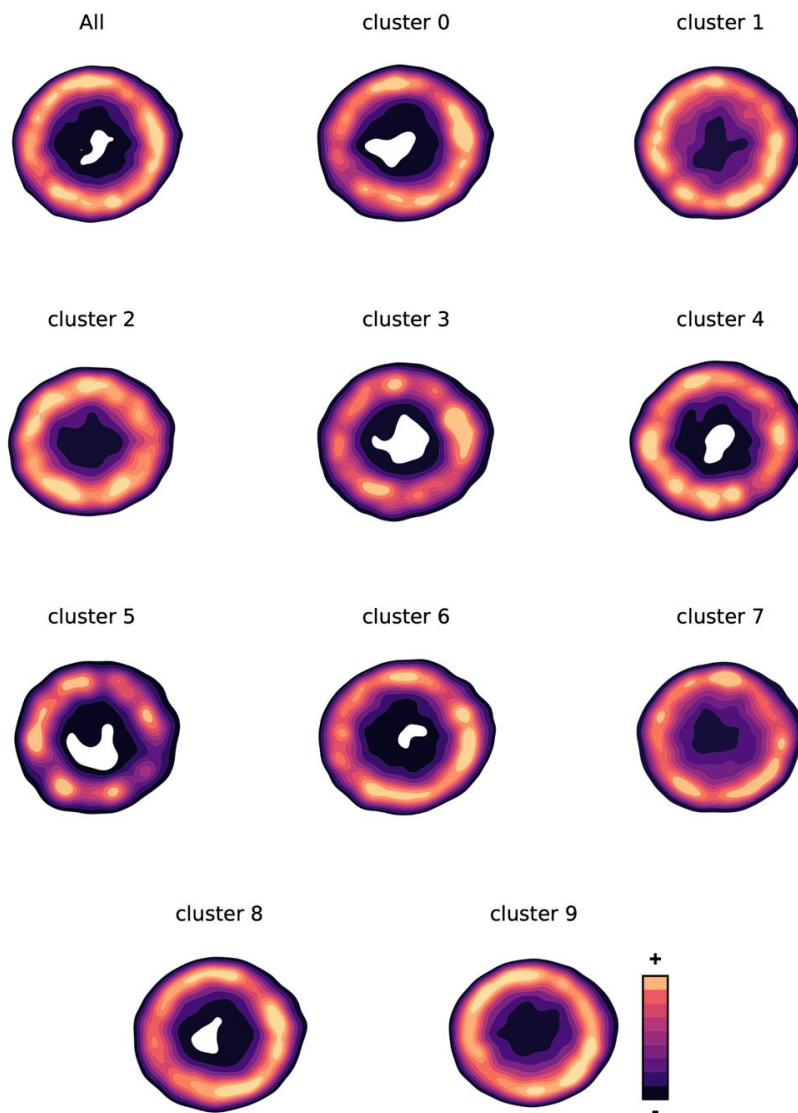
Supplemental figure 10. Cluster enrichment per experimental condition in the second to fourth optimal bins for the single-animal embeddings on the SI task. Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using the Benjamini-Hochberg method across both clusters and bins. In all cases, N=26 for NS and N=27 for CSDS. A) Second bin (126 to 252 seconds). Significant differences observed in clusters 3: $U=1.9e+2$, $p=6.3e-10$, 4: $U=5.9e+2$, $p=1.4e-5$, 7: $U=1.6e+2$, $p=6.9e-4$, and 8: $U=6.55e+2$, $p=6.3e-8$ B) Third bin (252 to 378 seconds). Significant differences observed in clusters 2: $U=1.8e+2$, $p=1.8e-3$, 3: $U=1.7e+2$, $p=1.2e-3$, 5: $U=4.9e+2$, $p=8.5e-3$, 6: $U=1.9e+2$, $p=7.01e-3$, 7: $U=1.7e+2$, $p=9.6e-4$, and 8: $U=6.3e+2$, $p=6.6e-7$. C) Fourth bin (378 to 504 seconds). Significant differences observed in clusters 4: $U=5.2e+2$, $p=2.5e-5$, and 8: $U=6.02e+2$, $p=6.5e-6$. Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. Source data are provided as a Source Data file.



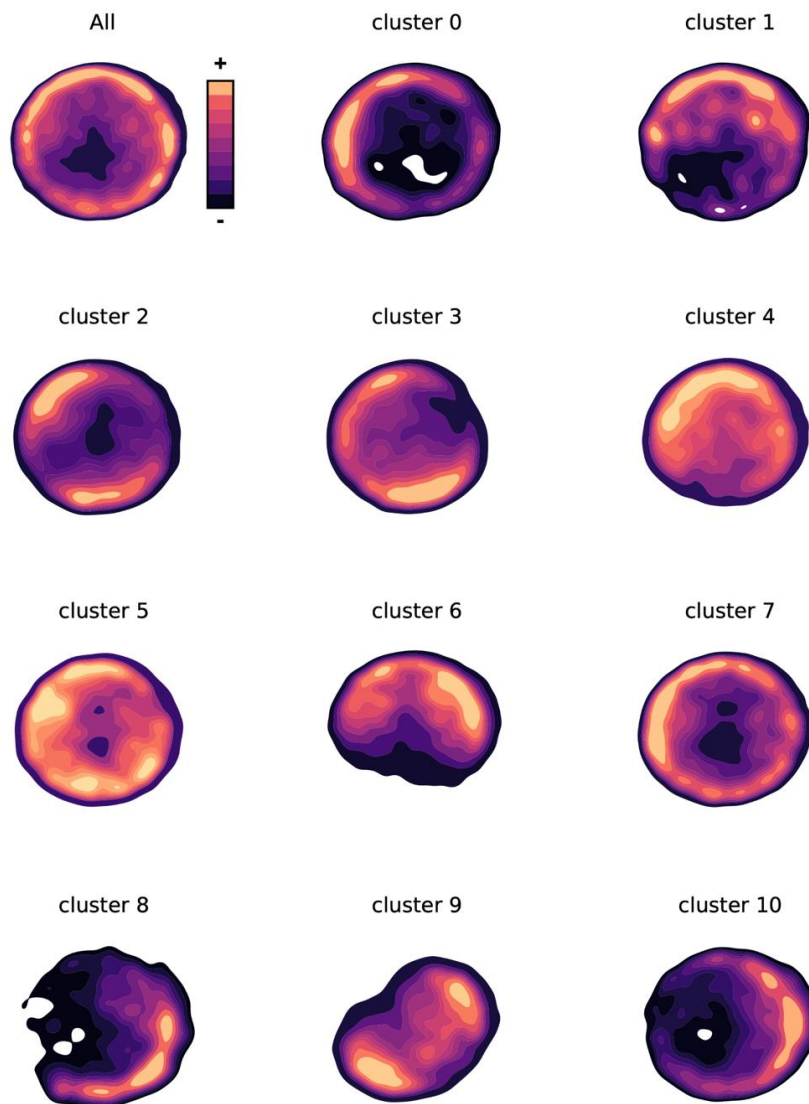
Supplemental figure 11. Cluster enrichment per experimental condition in the second to fourth optimal bins reported for the multi-animal embeddings on the SI task. Reported statistics correspond to a 2-way Mann-Whitney U non-parametric test corrected for multiple testing using the Benjamini-Hochberg method across both clusters and bins. In all cases, N=26 for NS and N=27 for CSDS. A) Second bin (124 to 248 seconds). No significant differences observed. B) Third bin (248 to 372 seconds). Significant differences were observed in clusters 0: $U=5.2e+2$, $p=3.3e-3$, and 5: $U=5.3e+2$, $p=1.6e-3$. C) Fourth bin (372 to 496 seconds). No significant differences were observed. Bar graphs represent mean \pm standard deviation of the time proportion spent on each cluster. Source data are provided as a Source Data file.



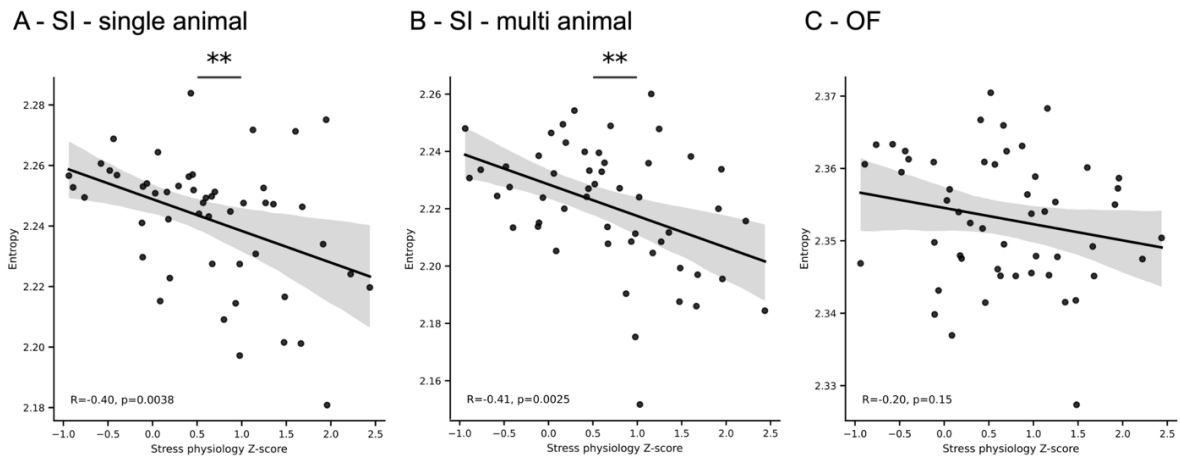
Supplemental figure 12. Spatial distribution of clusters obtained using single-animal embeddings in the SI task. Heatmaps include full trajectories of all experiments in both conditions, filtering time points belonging to each obtained cluster, and without filtering (labelled as "all"). White background indicates null population of the area. All clusters enriched in CSDS show lower occupation of the center of the arena than those enriched in NS animals.



Supplemental figure 13. Spatial distribution of clusters obtained using multi-animal embeddings in the SI task. Heatmaps include full trajectories of all experiments in both conditions, filtering time points belonging to each obtained cluster, and without filtering (labelled as "all"). White background indicates null population of the area. All clusters enriched in CSDS show lower occupation of the center of the arena than those enriched in NS animals.

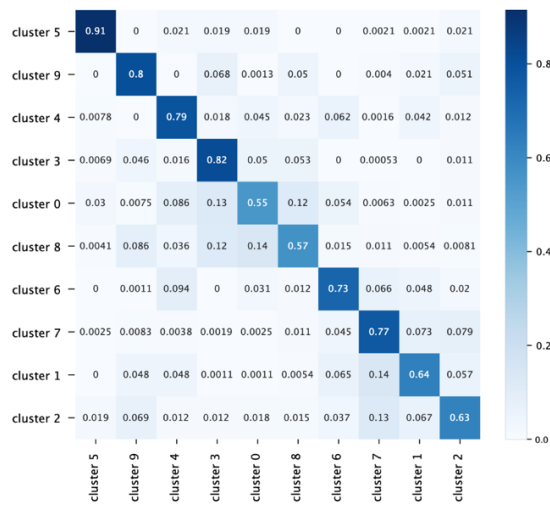


Supplemental figure 14. Spatial distribution of clusters obtained in the OF task. Heatmaps include full trajectories of all experiments in both conditions, filtering time points belonging to each obtained cluster, and without filtering (labelled as "all"). White background indicates null population of the area. All clusters enriched in CSDS show lower occupation of the center of the arena than those enriched in NS animals.

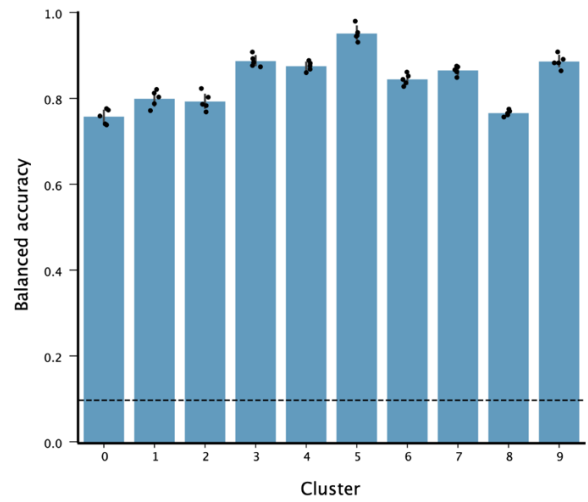


Supplemental Figure 15. Correlation between behavioral entropy and stress physiology Z-score. A) Behavioral entropy of the cluster space obtained with single animal embeddings during the social interaction (SI) task shows a significant negative Pearson correlation with the stress physiology Z-score ($R=-0.40$, $p=3.8e-3$, $N=53$). Error bands represent the 95% confidence band around the mean of the linear model. B) Behavioral entropy of the cluster space obtained with multi-animal embeddings during the social interaction (SI) task shows a significant negative Pearson correlation with the stress physiology Z-score ($R=-0.41$, $p=2.5e-3$, $N=53$). Error bands represent the 95% confidence band around the mean of the linear model C) Behavioral entropy of the cluster space obtained during the open field (OF) task shows no significant Pearson correlation with the stress physiology Z-score ($R=-0.20$, $p=0.15$, $N=53$). Error bands represent the 95% confidence band around the mean of the linear model. All three tests are two-sided. Source data are provided as a Source Data file.

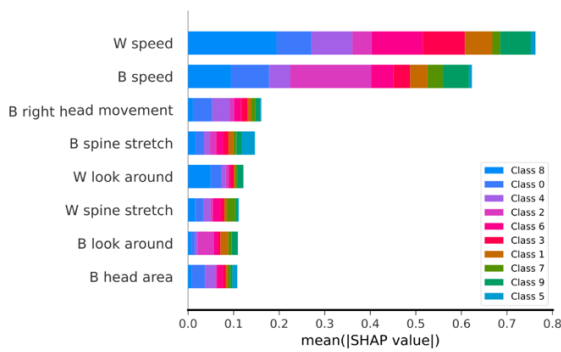
A - cluster detection confusion matrix



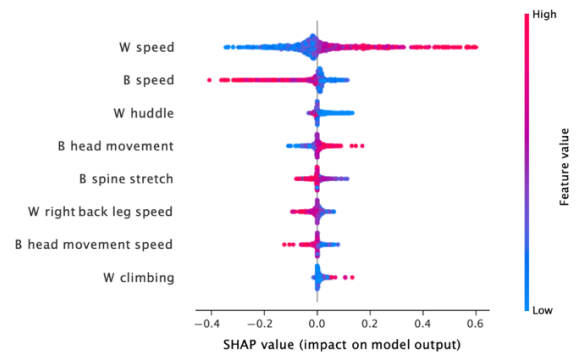
B - cluster detection performance



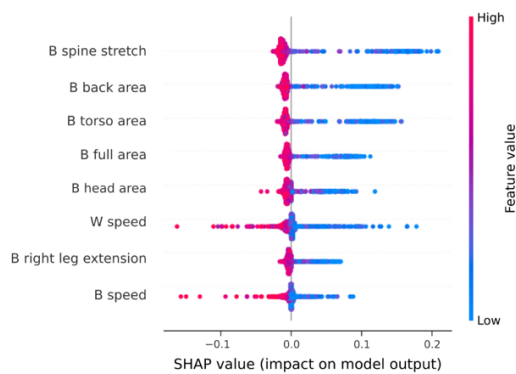
C - SHAP global feature importance



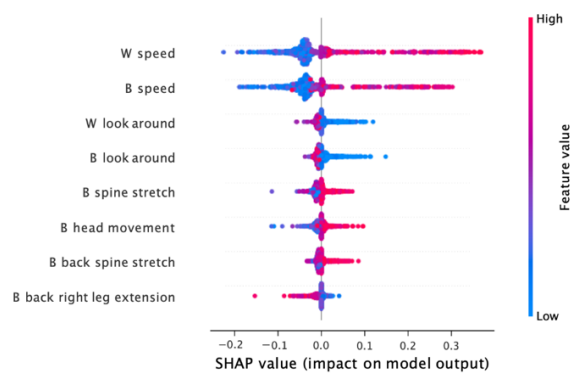
D - SHAP analysis of SI multi-animal cluster 3



E - SHAP analysis of SI multi-animal cluster 5



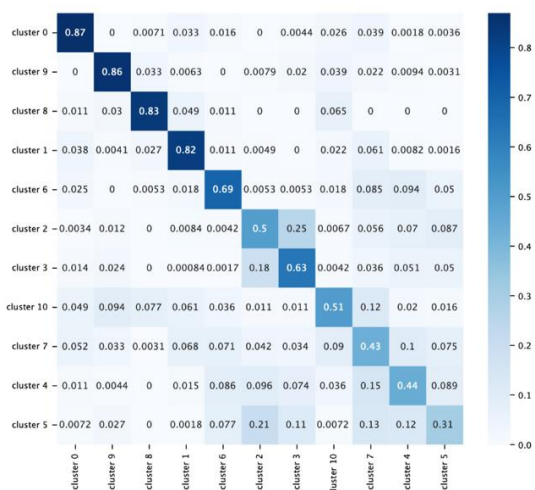
F - SHAP analysis of SI multi-animal cluster 9



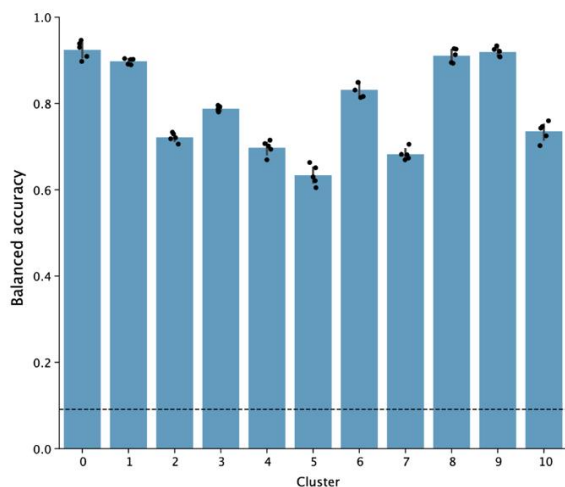
Supplemental Figure 16. SHAP analysis of unsupervised cluster assignments in the multi-animal social interaction task. Gradient boosting machines were trained to map from a predefined set of time series statistics (including body part speeds, distances, distance speeds, areas, area speeds, and supervised annotations for each of the two animals and their interaction) to the previously obtained cluster assignments. A) Confusion matrix obtained from the trained gradient boosting machine classifying

between clusters. Aggregated performance over the validation folds of a 5-fold cross-validation is shown. B) Validation performance per cluster across a 5-fold (N=5) cross-validation loop. Balanced accuracy was used to correct for cluster assignment imbalance. The dashed line marks the expected performance due to chance, considering all outputs. Bars show mean \pm 95% confidence interval. C) Overall feature importance for the multi-output classifier using SHAP. Features in the y-axis are sorted by overall absolute SHAP values across clusters. Classes on the bars are sorted by overall absolute SHAP values across features. D-F) Bee swarm plots for the three most differentially expressed clusters between NS and CSDS mice (3, 5, and 9), identified with the unsupervised DeepOF pipeline on the SI experiments using single-animal embeddings. The depicted plots display the first 8 most important features for each classifier, in terms of the mean absolute value of the SHAP values. Source data are provided as a Source Data file.

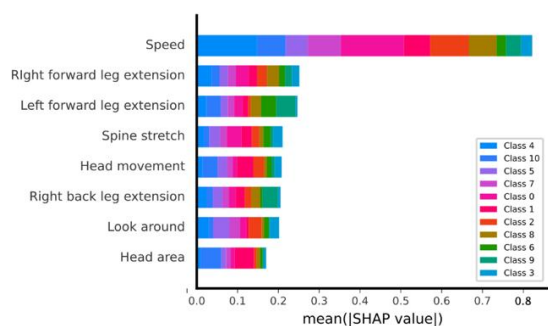
A - cluster detection confusion matrix



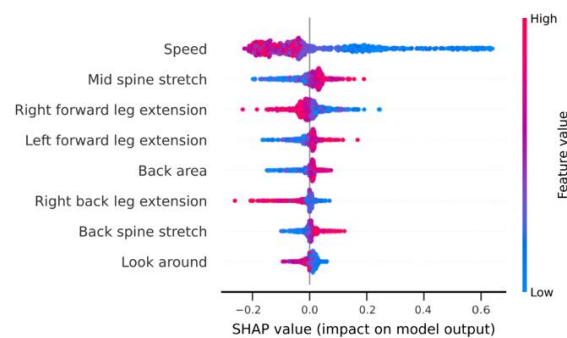
B - cluster detection performance



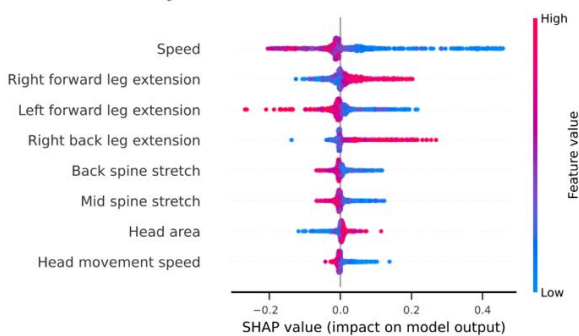
C - SHAP global feature importance



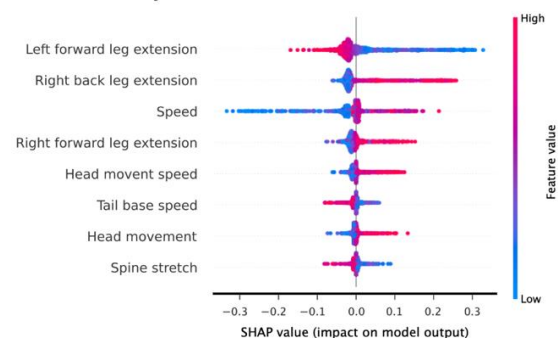
D - SHAP analysis of OF cluster 0



E - SHAP analysis of OF cluster 8



F - SHAP analysis of OF cluster 9



Supplemental Figure 17. SHAP analysis of unsupervised cluster assignments in the open field task. Gradient boosting machines were trained to map from a predefined set of time series statistics (including body part speeds, distances, distance speeds, areas, area speeds, and supervised annotations) to the previously obtained cluster assignments. A) Confusion matrix obtained from the trained gradient boosting machine classifying between clusters. Aggregated performance over the validation folds of a 5-fold cross-validation is shown. B) Validation performance per cluster across a 5-fold (N=5) cross-validation loop. Balanced accuracy was used to correct for cluster assignment imbalance. The dashed line marks the

expected performance due to chance, considering all outputs. Bars show mean \pm 95% confidence interval. C) Overall feature importance for the multi-output classifier using SHAP. Features in the y-axis are sorted by overall absolute SHAP values across clusters. Classes on the bars are sorted by overall absolute SHAP values across features. D-F) Bee swarm plots for the three most differentially expressed clusters between NS and CSDS mice (4, 9, and 10), identified with the unsupervised DeepOF pipeline on the SI experiments using single-animal embeddings. The depicted plots display the first 8 most important features for each classifier, in terms of the mean absolute value of the SHAP values. Source data are provided as a Source Data file.

6 Discussion

6.1 There and back again: towards systematic quantification of natural behavior

Understanding how living organisms (humans included), interact with and react to the environments they are exposed to has captured scientific curiosity since antiquity. As introduced in chapter 1, modern science has come a long way since then, and a plethora of approaches have been proposed, from observational ethological studies to extremely reductionist and question-specific settings.

The advent of machine-learning-based tracking and quantification approaches is particularly exciting because it evolves in a way that is applicable across the entire board. First off, automatic quantification methods pose a unique opportunity to systematize observational studies in the wild in non-invasive ways, even without human presence. This can have tremendous impact not only in our understanding of ethology itself, but also pave the way for innovations in ecology and wildlife conservation.

Along these lines and starting broad, the current rapid decline of animal diversity (genetic, ecological, and behavioral) underscores the urgent need for tools that can conduct swift and comprehensive assessments of wildlife diversity and population dynamics [144]. Traditional data collection methods, which rely heavily on human fieldwork, present numerous challenges including time and cost, potential threats to wildlife and human safety, and the inevitable generation of biased datasets. These limitations significantly hinder our understanding of global ecological dynamics and the effectiveness of our conservation efforts.

However dismal the landscape may look, technological advancements show some light at the end of the tunnel. Alongside hardware breakthroughs, such as trap or on-animal cameras and automatic drones, the advancements in freely available markerless pose estimation tools presented earlier in this thesis can help in a number of ways, such as individual identification, detection of migration patterns with static sensors, injury detection, and quantification of social dynamics in the wild, to name a few [65].

The potential of these technologies to enhance our understanding of animal ecology, streamline conservation efforts, and even illuminate new paths for wildlife preservation is enormous. The promise they hold, coupled with their integration with machine learning, could play a significant role in turning the tide on the alarming decline in animal diversity.

Moreover, the current thesis is an example of how the opposite trend can be executed: instead of relying on artificially simplistic models that enable simple quantification in laboratory settings, richer environments can be put in place without sacrificing rigor, by leveraging markerless pose estimation and automated quantification methods.

Thus and so, and after exploring the state of the art in chapter 2, chapters 3 and 4 introduced a novel open-source tool, called DeepOF, capable of examining both individual and social behavioral patterns in rodents using data annotated through DeepLabCut pose estimation, in supervised and unsupervised ways. Furthermore, chapter 5 delved into how, by applying this tool, we characterized unique individual and social behavioral profiles following CSDS, identified through traits annotated by DeepOF on C57Bl/6N subjects. Also, comparable results were obtained with our unsupervised pipeline, capa-

ble of recognizing behavioral shifts across various experimental contexts, including social interaction, single-animal open field tests, and social avoidance tasks. Furthermore, by exploring behavioral dynamics, DeepOF allowed us to systematically pinpoint how the initial moments of interaction with a new same-species partner are crucial for the social profiling of CSDS exposure in both supervised and unsupervised pipelines.

In this final chapter, we will delve into the impact that our tool can represent on the field it is immersed in, both in terms of technology development and knowledge discovery.

6.2 DeepOF in context: the current landscape of open-source software for behavioral analysis

The release of DeepLabCut in 2018 was arguably the cornerstone of a methodological revolution in the field of behavioral neuroscience. Since then, a plethora of tools have enabled researchers not only to quickly automate previously laborious manual quantification, but to think outside the box and design more complex and naturalistic new experimental settings altogether. While empowering, the current software landscape has quickly become little short of daunting: packages for pose estimation itself [61, 62], supervised annotation [71, 77], unsupervised analysis [78, 76, 80], and so on, propose constant innovation in a field that is still to stabilize to a new status quo, in a rapid turnover fashion that mimics the current state of other AI-dependent scenarios [145]. In this context, DeepOF offers the (to date) unique advantage of being an easy-to-use, label-free exploratory tool capable of annotating and analyzing motion-tracking data with just a few well-documented commands. This makes it easy for new users to adopt and try the software without big commitments, which we believe is key to success in such a rapidly changing field. Moreover, far from being a mere compilation of previously established methods, most algorithms presented are custom and adapted specifically to the tasks they are deployed to be used for. This way, easy adoption is contrasted with choice and customization, if a given user desires to take advantage of it. DeepOF is not designed to beat other available modules in their own game, but rather to act as a complement: after running an unsupervised pipeline and obtaining results that hint at particular (although non-pure) behaviors, a researcher could use the acquired knowledge to label and train supervised models using a tool like SimBA [71], for example, in order to confirm their suspicions.

Moreover, a significant advantage of DeepOF, SimBA [71], VAME [80], and many of the tools mentioned in this thesis is their open-source nature. Besides increasing transparency, which is always key to reproducibility, one must not forget that all these packages are being developed by (and mostly for) non-profit research organizations that thrive by interacting, debugging, and building on top of each other. This sort of synergetic, interdependent competition is a core aspect of modern science, and having access to code (including models, training schemes, and data) is crucial. Furthermore, a side effect of the overwhelming adoption of motion tracking software such as DeepLabCut is the increasing number of public datasets that are being released, which in turn enable not only the training of more powerful architectures that need less supervision [69], but

also the creation of open-source competitions and benchmarks. The Caltech Mouse Social Interactions (CalMS21) dataset, for example, is a pioneer in providing benchmarks for the detection of social interactions, annotation style transfer, and identification of rare traits [146]. Although unsupervised learning benchmarking is largely uncharted territory so far, it will be crucial to compare the DeepOF pipeline with other available methods in this area as the tools become accessible. Finally, although requiring specific hardware in many cases, such as increasingly powerful GPUs, all mentioned software is free to use, which makes it broadly accessible for research groups with limited resources. This is a huge advantage over the proprietary, often expensive, previous state-of-the-art [147].

Moreover, several extremely recent developments introduced significant progress on foundation models for markerless, one-shot video tracking. Efforts such as TAPIR (*Tracking Any Point with per-frame Initialization and temporal Refinement*, from *DeepMind*) [148] and *OmniMotion* [149] allow users to track any point in a given video upon labelling a single frame. While integration attempts into DeepOF have shown that tracking accuracy is yet to match DeepLabCut and other neuroscience-oriented programs, ease of use could lead to massive adoption of these pipelines as soon as models get better, with efforts in fine-tuning these general-purpose pretrained models to more specific tasks probably playing a big role in the near future.

A word of caution should be stated, however, since these models (as many others) have also increasing malicious potential if falling into the wrong hands: lightweight, powerful models for tracking and identifying individuals could be used for illegal surveillance, for example. As is currently the case in other fields, such as large language models (LLMs) [150, 151], I believe ethical considerations need to be thoroughly taken into account when open-sourcing, especially as datasets become larger and models more capable and easier to tune.

6.3 Perspectives on supervised learning on behavioral data

Going back to the supervised pipeline provided within DeepOF, we should highlight that it offers a set of rule-based annotators and pre-trained models that free the user from the need to manually label their data. While convenient and easy to use, this approach is extremely limited to simple behaviors that can be either reduced to simple but stable rules (such as nose-to-nose contacts or climbing behaviors) or robustly generalizable across datasets (such as the huddle classifier presented in chapter 5). With the increasing popularity of these tools in the research community and the aforementioned rapidly growing corpus of datasets and related competitions, it is to be expected that more complex traits will achieve similar transfer learning results in the near future. This would dramatically simplify the process of labeling and detecting specific, pre-defined behaviors, potentially eliminating the need for training new models altogether, in a fashion that would resemble the current discussion on foundation models [152].

Furthermore, the current developments in Large Language Models (LLMs) and text interfaces for image processing and generation [153, 154] suggest that a future where

describing specific, unlabeled behaviors with text to an LLM-video model capable of automatically recognizing patterns is (although far from the current state of the art) within reach, and an interesting path forward.

6.4 Perspectives on unsupervised learning on behavioral data

As introduced in chapter 1, even if adopting a purely mechanistic definition, behavior is not inherently discrete, but arguably hierarchical. Complex actions can always be decomposed into simpler ones (typically referred to as primitives) whose repetitive nature makes them simpler to detect. Discretizing motion tracking data is therefore (as is often the case in other fields too) an ill-defined problem: with no natural solution present, a given set of clusters will focus on some aspects of behavior, leaving others behind. This renders discretization a utility problem that serves the purpose of allowing researchers to test hypotheses related to a broader scope, but which is arguably only secondary to learning robust representations.

While in DeepOF we have focused primarily on learning useful discretization models of motion, I believe future work in this direction should focus on understanding the underlying learned representations better. Along these lines, the field of representation learning has set the core principles a good and robust representation should be able to follow [120]. First, representations should be **expressive**, in the sense that they should be able to represent an exponential amount of configurations for their size. This would contrast, for example, with other representations such as one-hot encodings or mere hard cluster assignments. Second, good representations should be **robust** to small and local variations in input data. As an example, behavioral representations in DeepOF should vary neither with the position of the animals in the arena nor with their rotational orientation, hence these sources of variance are removed during processing. Third, good representations should be **disentangled**, meaning that learned dimensions should be uncorrelated and represent distinguishable factors with identifiable meanings. Despite not being perfect, this set of principles serves as a rule of thumb to design interpretability tests and analyses, and their usefulness in this context remains to be explored.

This type of analysis, together with the development of systematic benchmarks for unsupervised learning on motion tracking, would be an ideal framework to formally compare all the models provided within DeepOF. So far, comparisons were purely functional to choose a good default for the deployed package while exploring different variants that extended the state of the art in the field. Thus, metrics such as training time and compute resources needed, and discrimination capabilities between global animal embeddings across experimental conditions (as presented in chapter 5), were the main model selection criteria. The exception is the introduction of contrastive learning models, which were included in the package after the submission of the paper, and yielded comparable results with shorter training times and fewer parameters. This is exciting news moving forward and sets self-supervised alternatives as the most likely course for further immediate model development.

Thus and so, unsupervised (and self-supervised) representation learning evaluation remains, in my opinion, the most critical point for research in the immediate future. This would not only enable robust hypothesis testing and latent manipulation, but also alignment across modalities, as will be explored in the next sections.

6.5 Increasing resolution in neurobiological research: behavioral quantification in context

A living system is far more than the sum of its parts, with different biological levels interacting and regulating one another constantly in complex ways. From genetics, transcriptomics, epigenetics, and proteomics, to neural signaling, behavior, and environmental factors, being able to capture information from different biological levels in clever ways can be key to understanding any phenotype [68].

Along these lines, the presented breakthroughs in motion tracking are not isolated. Recent years have seen remarkable progress in other areas relevant for neurobiological research, following a common trend of increasing experimental resolution, also outside the temporal domain [68]. Interestingly, in many fields other than motion tracking, breakthrough developments came mostly from the hardware side, which in turn allowed researchers to collect more data and raised the stakes of data analysis and software development in specific domains.

As a representative example, we can explore the field of transcriptomics, where developments in single-cell resolution sequencing technologies sparked a plethora of tools and methods that innovate how data are analyzed. Here, programs like SCANPY [155] and SEURAT [156] have earned recognition as the state of the art in the field, providing high-quality sets of tools, workflows, benchmarks, tutorials, and user support. They also have grown an extensive user community, which creates feedback loops with contributions and extensions which improve the software constantly. A flagrant example of such an extension is SquidPy [157], a package that focuses on the forefront of spatial transcriptomics, which is greatly helping to improve our understanding of how cells in tissues are organized and interact with each other.

All in all, this standardization offers many benefits for several connected fields (including stress research, as explored in our published commentary on STRESS [68]), and has a big impact not only on our basic understanding of cell makeup and gene activity in key tissues, but also on the discovery of new drug targets and development of new treatments. As a concrete example, in 2022 Lopez et al. used a mix of automatic behavior tracking methods and single-cell RNA-sequencing techniques to discover specific molecular patterns in different stress-related cell types, and reported a new way in which the long-lasting antidepressant effects of ketamine work in a certain type of nerve cell in a specific part of adult mice's brains [102]. In this paper, motion tracking technology is used to automatically assess shifts in behavior across different experimental groups, illustrating how automated behavioral drug screenings can be carried out.

Moreover, transcriptomics is far from being the only case. Proteomics, for example, is being positively impacted by new developments in techniques such as mass spectrometry

[158], and by AI-powered tools such as *AlphaFold* [159], which are solving biochemical problems that until a few years ago were deemed unreachable. Brain imaging and neural activity measuring are other relevant fields that have seen recent relevant advances, such as resolution improvements for functional MRI [160], and real time joint behavioral-motion capture using miniscopes and calcium sensors [161, 162].

As illustrated in the aforementioned paper, the combination of motion tracking quantification with many of these tools holds a lot of potential to measure the impact of genetic and biochemical changes in behavior systematically. However, these techniques often describe different (albeit non-orthogonal) axes of the same phenomena. Although learning from and interpreting each on its own can already be useful to expand our knowledge in many ways, much information is lost in the process.

6.6 Beyond motion tracking: integrating multimodal data

The science of learning how to better integrate these different sources of data, which can lead to a more holistic understanding of the underlying, common phenomena under study, is often referred to as **multimodal integration** (or **multimodal learning**, in the context of ML). As a side note, it is important to highlight that behavior itself is more than motion alone, and adopting a broad definition would require integrating additional variables that cannot (at least to date) be extracted from video. These include, for example, things like heart rate, respiratory rate, vocalization, and neural activity.

At a basic level, multimodal integration thus requires researchers to draw conclusions from experiments describing multiple (complementary) axes of the same problem and drawing conclusions explaining all observed patterns. While a naïve approach may be to align the raw variables themselves (over time, for example, in the case of behavior, or as concatenated input to a model, in what is called **early integration**), there are several inherent problems that would need to be solved. For starters, different data modalities may rely on different hardware, with different collection rates, artifacts that would need to be removed, sensitivities, and overall limitations [163]. This renders raw data alignment extremely hard, and forces researchers to often analyze modalities separately and draw joint conclusions manually. This separate processing is often called **late integration**, and while it solves many of the presented limitations, it carries the strong disadvantage of disregarding joint distributions across modalities, focusing exclusively on the marginals. When modalities are uncorrelated, however, this can be an extremely powerful framework, as illustrated by multimodal ensemble learning [164].

Building on the previous section, the main focus of current approaches to multimodal integration deals with *aligning data representations* (in what is known as **middle integration**) instead of the data themselves. By learning robust representations that extract features invariant to hardware noise and timescale nuances, these approaches hold the promise of having the best of both worlds: good alignment, while retaining and learning joint distributions. Thus and so, self-supervised approaches such as those presented in chapter 3 are showing promising results, since they allow several crucial levels of flexibility, such as having modality-specific encoders (which can deal with different

types of data naturally), and specifically crafted contrastive positive and negative sampling schemes. Along these lines, the recently published package CEBRA [165] offers a representation learning framework to learn joint embeddings using motion tracking and neural activity data with contrastive approaches. By aligning both modalities at the embedding level, CEBRA is capable of reporting non-linear neural correlates of motion, directly enabling questions regarding how one affects the other in complex ways that may be difficult to detect without computer assistance. Moreover, and in line with what was explored before in this section, two main positive and negative sampling schemes are provided: a purely unsupervised one, based on time alone and similar to that presented for DeepOF in chapter 3, and a supervised one based on annotated labels. This makes it easier for researchers to choose between a more exploratory embedding of neural-motion interactions, or a hypothesis-driven one that can answer specific questions.

All in all, as both hardware and software technology advance, new methods are being developed that enable researchers to get a more holistic view of living organisms as systems, instead of independent collections of unrelated features. As time advances, I expect these approaches to become more prevalent and lead to better representations. While multimodal integration holds an exciting and extremely useful potential for research moving forward, however, motion tracking has the advantage of relying on relatively affordable hardware (video cameras) which enabled its wide adoption in the first place. It should thus not escape our attention that including more data modalities can be prohibitively hard, both in terms of labor intensity for data collection and elevated costs, especially in resource-constrained labs. This renders parallel efforts in representation learning on motion tracking data alone (such as DeepOF) extremely relevant too.

6.7 Impact of the presented results in chronic stress research

With the deeper understanding of the current status of behavioral analysis (and motion tracking in particular) built over the last few sections, we can now explore the impact of the presented research in our understanding of the model introduced as a case study: Chronic Social Defeat Stress. As explored in chapters 1 and 5, the individual and social behavior of animals exposed to CSDS has been extensively researched using models such as elevated plus mazes and social avoidance tasks, which can distinguish anxiety-like and altered social behaviors between cases and controls. This thesis has displayed several ways in which DeepOF has improved the state of the art in this regard, both reducing experimental effort and enabling greater analysis detail.

For starters, the observation that after exposure to the aggressive conspecific during the CSDS pipeline, experimental subjects' behavior follows an arousal pattern that fades over time due to habituation is, to the best of our knowledge, novel. The first relevant contribution of this thesis to CSDS research is then the supervised and unsupervised quantification of this arousal period, which in all our datasets was between two and two and a half minutes (and therefore close to the typical duration of a social avoidance task [98]). These results were moreover absent in single animal settings, which further supports this interpretation.

6.7 Impact of the presented results in chronic stress research

In line with these findings, our study showcases how the behavioral annotation provided within DeepOF leads to a more effective distinction of the social behavioral profile between stressed and non-stressed animals compared to the traditional SA task. This is a non-trivial finding: while capable of more detail, DeepOF is a general purpose tool, whereas the aforementioned task was specifically designed to detect this phenotype. I believe this is a great example of how overfitting specific measurements to our experimental designs may not always be optimal, and of how carefully tested exploratory tools can take us extremely far already.

Another result worth revisiting in this section is the differential entropy between stressed and non-stressed animals reported from our unsupervised pipeline. The fact that stressed animals display lower entropy in the discrete behaviors they explore is also non-trivial, and it highlights how reduced and focused on avoiding a potential stressor behavior becomes in stressful situations, arguably in line with the fight-or-flight response [166].

Besides these specific contributions, DeepOF holds potential to explore in even more depth this model, such as for the identification of animals susceptible to stress and those resilient to it, which are often determined using oversimplified outcomes in the aforementioned social avoidance test, such as the fraction of time experimental animals spent close to their conspecific. While this variable (known in the literature as SA ratio) effectively distinguishes individuals affected by stress, especially in more severe CSDS conditions, our approach seems to significantly enhance the scope and sensitivity of this distinction, although more research is needed in this regard. Moreover, a tool included in DeepOF that was not put to use in this thesis and can work well in this context is the possibility to train control normative models. These work by fitting Gaussian kernel densities to the global animal embeddings of control animals, and reporting differences in the likelihood under the model between conditions. Stressed animals with embeddings that are closer to the control population could then be tagged as resilient. The next and final section on translational research applications will further explore this idea, showcasing its potential for more complex settings, such as the detection of the depression-like syndrome presented in chapter 1.

In conclusion, the annotation pipelines implemented in DeepOF provide a more comprehensive and precise individual and social behavioral profile of animals exposed to CSDS when compared to the previous state of the art in the field. This has several implications moving forward, such as the potential adoption of DeepOF (or similar tools) for CSDS quantification as a standard procedure, and the deeper exploration of the tool for other aspects not discussed here. Moreover, an important factor contributing to the overall success of DeepOF in the presented social behavioral profiling lies in its experimental setup. While the social avoidance task relies on confined animals (typically in wired mesh cages, which prevent natural interaction between freely moving animals), open field settings allow for a much more natural interaction. Moreover, in the SA task, confined animals may display anxiety-related behaviors that influence their physiological state and their social interaction and approach behaviors with the conspecific.

Finally, and while we believe that our contributions to CSDS are significant and worth mentioning, we should not lose sight of the broader scope. Chronic stress is just one

example setting in which such pipeline can be applied, and much more remains to be explored in other equivalent or more complex models. The next and final section will explore this in detail, focusing on how translational research can be positively affected with tools such as these.

6.8 Frontiers of the field: between translational research and knowledge discovery

As thoroughly explored in chapter 1, the current state of research in applied neurobiology and psychiatry research lies far from the clinic. While many studies present innovations in drug development, genetic markers, and more, little is translated to real patients, in a phenomenon that has been described as the translational gap [90]. Moreover, despite animal models being adopted decades ago, their use to mimic mental disorders hasn't managed to live up to the expectations. This is due to many reasons, such as the complexity of mental disorders per se and their prominent environmental causes, which are to date difficult to replicate in animals [31]. On top of this, and the lack of biologically-driven definitions of mental disorders makes the problem harder, as currently described entities could correspond to more than one etiologically relevant entity [27].

Even when taking all these issues into account, I believe there is light at the end of the tunnel, and that the potential of modern behavioral quantification in this regard is significant. Firstly, because it allows for finer-grain measurements that can be used to get more disentangled and data-driven definitions of the diseases under study, as seen with the RDoC initiative [28]. Second, because once these definitions are agreed upon, this technology could simplify the accurate assignment of labels to subjects under study, decreasing the focus on more subjective measurements [27]. Moreover, as seen with the depression-like syndrome introduced in chapter 1, these disentangled definitions are key to improving back-translation [31]. That is, the definition of human-equivalent diseases in animal models that are as close as possible to the clinically relevant phenotype. In this regard, the normative modelling pipeline introduced in the previous section can play a key role: as a follow-up study to what was presented in chapter 5, we are currently using DeepOF to build domain specific normative models for DLS. This way, animals can be scored on each behavioral domain that escapes the species barrier (which are *loss of energy and fatigue, lack of concentration and indecisiveness, psychomotor agitation or retardation, disturbed sleep with hyper or hyposomnia, appetite or weight changes, and diminished interest or pleasure in activities*). By detecting shifts on each of these domains, many of which are carried out with DeepOF, we can get individual profiles for each experimental mouse. This way, DeepOF can be used in two stages, first to select individuals that meet stricter inclusion criteria for follow-up studies, and second to detect shifts in behavior upon applying a treatment (such as a drug).

Moreover, these technologies are not limited to animal models. Given that the ultimate goal of clinical research is to comprehend and enhance the quality of life for humans, assigning humans to the correct labels, and study their shifts in behavior systematically, is also crucial. In this context, advancements in comprehending human behavior

through virtual reality (VR) are noteworthy. Presently, VR enables researchers to accurately monitor movement in meticulously designed settings, facilitating the transfer of paradigms like fear conditioning to human participants noninvasively [95, 81]. Tools such as DeepOF can be applied to this type of data as well [134].

Finally, and while translation to the clinic is set as the final goal in this context, these technologies can also be of great help to acquire new knowledge. For example, the use of unsupervised learning in motion tracking data has the potential to uncover new behaviors that are systematically expressed in certain conditions, although more research in novel situations is needed in this regard [34]. Moreover, detecting unsupervised shifts in behavior can be of great use in many high-throughput and hypothesis-free situations, such as Quantitative Trait Loci (QTL) mapping [167]. This refers to a statistical method that aims to link two types of information, namely phenotypic data (quantitative traits, as behavior in this case) and genotypic data in the cohorts under study. This way, researchers can identify regions in the genome that can influence the variation of a given trait.

Even though in the context of DeepOF and similar tools these quantitative traits could be any measured parameter (such as speed, locomotion, social interactions, etc.), I believe the unsupervised animal embeddings introduced earlier are the most interesting opportunity in this regard. By detecting global shifts in behavior that are not associated with any particular hypothesis, researchers can increase throughput and scope massively. Moreover, detected shifts can then be analyzed individually, to dissect the differentially expressed patterns in a hypothesis-driven manner with the same tool. A similar idea is already being applied (although relying on supervised learning models detecting specific traits) for high-throughput drug discovery [168].

Linking together everything discussed in this section, DeepOF and similar tools are already being used today to revolutionize research in animal and human behavior, and hold increasing potential to have a positive impact on the current definitions of psychiatric conditions, improve pre-clinical and clinical trials, and aid relevant biological and drug discovery. The future of the field looks increasingly promising, and our contribution is but a grain of sand.

7 Conclusion & Outlook

7 Conclusion & Outlook

This thesis delved into the state of the art of animal motion tracking using key point estimation, and its further analysis using different annotation approaches. Three main goals were defined, including the implementation of novel deep clustering algorithms for the unsupervised analysis of motion tracking data, their deployment alongside other tools as part of a Python package, and their application to characterize a real-world behavioral model.

Along these lines, chapter 1 introduced the broad topics under discussion, including definitions of behavior, history of its quantification and application, and chronic stress as a case study. Moreover, it explored the broad technical foundations for what the thesis aimed to present.

Chapter 2 explored the technical aspects of behavioral analysis in more detail and presented state-of-the-art tools to analyze motion-tracking data in both supervised and unsupervised ways.

Subsequently, chapter 3 explored the methodological aspects of the newly introduced algorithms, and all analyses that were carried out for the presented results.

Chapter 4 then moved to introduce DeepOF, our novel Python package, as published in the *Journal of Open Source Software (JOSS)*. This paper, although short, serves as an entry point to the DeepOF ecosystem, its documentation, and contribution guidelines. Moreover, this publication included a code peer-review process of vital importance to what this thesis aims to stand for: open science, both in terms of methods and code. We believe this small paper is an important milestone for our vision of what DeepOF and similar tools represent in the field.

Next, chapter 5 demonstrated how the developed tools can be applied to a real world animal model, such as Chronic Social Defeat Stress. This set of results, published in *Nature Communications*, explores both supervised and unsupervised pipelines included, and how they yield overlapping yet complementary insights into the shifts in behavior that chronic stress causes in male laboratory mice. We hope this is but a kick-start example of what can be accomplished with this tool, and expect to gain insight into other animal models in the future, both with experiments carried out by direct colleagues and external users.

Finally, chapter 6 attempts to put the presented developments in context, delving into the potential impact of the provided tools in several related fields, such as ecology, integration with other data modalities, and QTL discovery. Moreover, a perspective on how the field is likely to evolve in the near future is discussed.

All in all, the current thesis is but an example of an evolving field, in which (as in many other areas currently powered by machine learning) novel tools are enabling both automation and discovery in ways that were not thought possible a decade ago. When putting these developments in context and as both technology and biological knowledge progress, from single cells to social behavior [68], the dream of jointly mapping behavioral responses to stimuli in a holistic manner is closer than ever.

Bibliography

- [1] D. A. Levitis, W. Z. Lidicker, and G. Freund. Behavioural biologists don't agree on what constitutes behaviour. *Anim Behav*, 78(1):103–110, 2009. doi:10.1016/j.anbehav.2009.03.018.
- [2] E. Charney. Genes, behavior, and behavior genetics. *Wiley interdisciplinary reviews. Cognitive science*, 8(1-2), 1 2017. URL: <https://pubmed.ncbi.nlm.nih.gov/27906529/>, doi:10.1002/WCS.1405.
- [3] F. Zhou, J. Ren, X. Lu, S. Ma, and C. Wu. Gene-Environment Interaction: A Variable Selection Perspective. *Methods in molecular biology (Clifton, N.J.)*, 2212:191–223, 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/33733358/>, doi:10.1007/978-1-0716-0947-7{_}13.
- [4] J. Cuevas. Neurotransmitters and Their Life Cycle. *Reference Module in Biomedical Sciences*, 1 2019. doi:10.1016/B978-0-12-801238-3.11318-2.
- [5] L. Luo. Principles of Neurobiology. *CRC Press*, 9 2020.
- [6] D. M. Gardner, David G; Shoback. *Greenspan's basic and clinical endocrinology, tenth edition*. McGraw-Hill Education, 10 2017.
- [7] B. S. McEwen. Hormones and behavior and the integration of brain-body science. *Hormones and behavior*, 119, 3 2020. URL: <https://pubmed.ncbi.nlm.nih.gov/31790663/>, doi:10.1016/J.YHBEH.2019.104619.
- [8] D. B. O'Connor, N. Gartland, and R. C. O'Connor. Stress, cortisol and suicide risk. *International review of neurobiology*, 152:101–130, 1 2020. URL: <https://pubmed.ncbi.nlm.nih.gov/32450993/>, doi:10.1016/BS.IRN.2019.11.006.
- [9] O. J. Bosch and L. J. Young. Oxytocin and Social Relationships: From Attachment to Bond Disruption. *Current topics in behavioral neurosciences*, 35:97–117, 2018. URL: <https://pubmed.ncbi.nlm.nih.gov/28812266/>, doi:10.1007/7854{_}2017{_}10.
- [10] American Psychiatric Association. Diagnostic and Statistical Manual of Mental Disorders. *Diagnostic and Statistical Manual of Mental Disorders*, 3 2022. doi:10.1176/APPI.BOOKS.9780890425787.
- [11] M. Fornaro, N. Clementi, and P. Fornaro. Medicine and psychiatry in Western culture: Ancient Greek myths and modern prejudices. *Annals of General Psychiatry*,

BIBLIOGRAPHY

- 8(1):21, 10 2009. URL: <https://annals-general-psychiatry.biomedcentral.com/articles/10.1186/1744-859X-8-21>, doi:10.1186/1744-859X-8-21/METRICS.
- [12] K. S. Kendler, K. Tabb, and J. Wright. The Emergence of Psychiatry: 1650-1850. *The American journal of psychiatry*, 179(5):329–335, 5 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/35331024/>, doi:10.1176/APPI.AJP.21060614.
- [13] E. R. Wallace and J. Gach. *History of Psychiatry and Medical Psychology*. Springer, 2008.
- [14] D. Pick. *Psychoanalysis: A Very Short Introduction*. OUP Oxford, 2015.
- [15] J. G. Holland. Behaviorism: part of the problem or part of the solution. *Journal of applied behavior analysis*, 11(1):163–174, 1978. URL: <https://pubmed.ncbi.nlm.nih.gov/649524/>, doi:10.1901/JABA.1978.11-163.
- [16] A. J. Rothschild. Psychopharmacology: Back to Basics. *Journal of clinical psychopharmacology*, 42(1):1–2, 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/34928555/>, doi:10.1097/JCP.0000000000001518.
- [17] C. J. Harmer, R. S. Duman, and P. J. Cowen. How do antidepressants work? New perspectives for refining future treatment approaches. *The lancet. Psychiatry*, 4(5):409–418, 5 2017. URL: <https://pubmed.ncbi.nlm.nih.gov/28153641/>, doi:10.1016/S2215-0366(17)30015-9.
- [18] I. R. Winship, S. M. Dursun, G. B. Baker, P. A. Balista, L. Kandratavicius, J. P. Maia-de Oliveira, J. Hallak, and J. G. Howland. An Overview of Animal Models Related to Schizophrenia. *Canadian journal of psychiatry. Revue canadienne de psychiatrie*, 64(1):5–17, 1 2019. URL: <https://pubmed.ncbi.nlm.nih.gov/29742910/>, doi:10.1177/0706743718773728.
- [19] M. Varghese, N. Keshav, S. Jacot-Descombes, T. Warda, B. Wicinski, D. L. Dickstein, H. Harony-Nicolas, S. De Rubeis, E. Drapeau, J. D. Buxbaum, and P. R. Hof. Autism spectrum disorder: neuropathology and animal models. *Acta neuropathologica*, 134(4):537–566, 10 2017. URL: <https://pubmed.ncbi.nlm.nih.gov/28584888/>, doi:10.1007/S00401-017-1736-4.
- [20] Q. Wang, M. A. Timberlake, K. Prall, and Y. Dwivedi. The recent progress in animal models of depression. *Progress in neuro-psychopharmacology & biological psychiatry*, 77:99–109, 7 2017. URL: <https://pubmed.ncbi.nlm.nih.gov/28396255/>, doi:10.1016/J.PNPBP.2017.04.008.
- [21] A. C. Campos, M. V. Fogaça, D. C. Aguiar, and F. S. Guimarães. Animal models of anxiety disorders and stress. *Revista brasileira de psiquiatria (Sao Paulo, Brazil : 1999)*, 35 Suppl 2(SUPPL.2), 2013. URL: <https://pubmed.ncbi.nlm.nih.gov/24271222/>, doi:10.1590/1516-4446-2013-1139.

- [22] G. Richter-Levin, O. Stork, and M. V. Schmidt. Animal models of PTSD: a challenge to be met. *Molecular psychiatry*, 24(8):1135–1156, 8 2019. URL: <https://pubmed.ncbi.nlm.nih.gov/30816289/>, doi:10.1038/S41380-018-0272-5.
- [23] M. Baker, S.-I. Hong, S. Kang, and D.-S. Choi. Rodent models for psychiatric disorders: problems and promises. *Laboratory Animal Research 2020 36:1*, 36(1):1–10, 4 2020. URL: <https://labanimres.biomedcentral.com/articles/10.1186/s42826-020-00039-z>, doi:10.1186/S42826-020-00039-Z.
- [24] R. A. Wise and C. J. Jordan. Dopamine, behavior, and addiction. *Journal of biomedical science*, 28(1), 12 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/34852810/>, doi:10.1186/S12929-021-00779-7.
- [25] D. O. Borroto-Escuela, P. Ambrogini, B. Chruścicka, M. Lindskog, M. Crespo-Ramirez, J. C. Hernández-Mondragón, M. P. de la Mora, H. Schellekens, and K. Fuxe. The Role of Central Serotonin Neurons and 5-HT Heteroreceptor Complexes in the Pathophysiology of Depression: A Historical Perspective and Future Prospects. *International journal of molecular sciences*, 22(4):1–13, 2 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/33672070/>, doi:10.3390/IJMS22041927.
- [26] A. Gururajan, A. Reif, J. F. Cryan, and D. A. Slattery. The future of rodent models in depression research. *Nature Reviews Neuroscience 2019 20:11*, 20(11):686–701, 10 2019. URL: <https://www.nature.com/articles/s41583-019-0221-6>, doi:10.1038/s41583-019-0221-6.
- [27] L. Miranda, R. Paul, B. Pütz, N. Koutsouleris, and B. Müller-Myhsok. Systematic Review of Functional MRI Applications for Psychiatric Disease Subtyping. *Frontiers in Psychiatry*, 12:1712, 10 2021. doi:10.3389/FPSYT.2021.665536/BIBTEX.
- [28] A. Vilar, V. Pérez-Sola, M. J. Blasco, E. Pérez-Gallo, L. Ballester Coma, S. Batlle Vila, J. Alonso, A. Serrano-Blanco, and C. G. Forero. Translational research in psychiatry: The Research Domain Criteria Project (RDoC). *Revista de psiquiatria y salud mental*, 12(3):187–195, 7 2019. URL: <https://pubmed.ncbi.nlm.nih.gov/29941228/>, doi:10.1016/J.RPSM.2018.04.002.
- [29] T. L. Bale, T. Abel, H. Akil, W. A. Carlezon, B. Moghaddam, E. J. Nestler, K. J. Ressler, and S. M. Thompson. The critical importance of basic animal research for neuropsychiatric disorders. *Neuropsychopharmacology 2019 44:8*, 44(8):1349–1353, 5 2019. URL: <https://www.nature.com/articles/s41386-019-0405-9>, doi:10.1038/s41386-019-0405-9.
- [30] J. Söderlund and M. Lindskog. Relevance of Rodent Models of Depression in Clinical Practice: Can We Overcome the Obstacles in Translational Neuropsychiatry? *International Journal of Neuropsychopharmacology*, 21(7):668–676, 7 2018. URL: <https://academic.oup.com/ijnp/article/21/7/668/4982723>, doi:10.1093/IJNP/PYY037.

BIBLIOGRAPHY

- [31] I. A. von Mücke-Heim, L. Urbina-Treviño, J. Bordes, C. Ries, M. V. Schmidt, and J. M. Deussing. Introducing a depression-like syndrome for translational neuropsychiatry: a plea for taxonomical validity and improved comparability between humans and mice. *Molecular Psychiatry* 2022 28:1, 28(1):329–340, 9 2022. URL: <https://www.nature.com/articles/s41380-022-01762-w>, doi:10.1038/s41380-022-01762-w.
- [32] R. W. Burkhardt. Commentary: New Directions in the History of Ethology. *Berichte zur Wissenschaftsgeschichte*, 45(1-2):189–199, 6 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/35680615/>, doi:10.1002/BEWI.202280103.
- [33] J. P. Scott. Comparative psychology and ethology. *Annual review of psychology*, 18:65–86, 1967. URL: <https://pubmed.ncbi.nlm.nih.gov/5333431/>, doi:10.1146/ANNUREV.PS.18.020167.000433.
- [34] A. Mathis, S. Schneider, J. Lauer, and M. W. Mathis. A Primer on Motion Capture with Deep Learning: Principles, Pitfalls, and Perspectives. *Neuron*, 108(1):44–65, 10 2020. URL: [http://www.cell.com/article/S0896627320307170/fulltexthttp://www.cell.com/article/S0896627320307170/abstracthttps://www.cell.com/neuron/abstract/S0896-6273\(20\)30717-0](http://www.cell.com/article/S0896627320307170/fulltexthttp://www.cell.com/article/S0896627320307170/abstracthttps://www.cell.com/neuron/abstract/S0896-6273(20)30717-0), doi:10.1016/J.NEURON.2020.09.017/ATTACHMENT/46E265F6-9FF2-4F64-9116-B8CC9C958FC4/MMC1.MP4.
- [35] Charles Darwin. *On the origin of species*. John Murray, 1859.
- [36] Charles Darwin. *The expression of emotions in man and animals*. John Murray, 1872.
- [37] P. Marler. Ethology and the origins of behavioral endocrinology. *Hormones and behavior*, 47(4):493–502, 2005. URL: <https://pubmed.ncbi.nlm.nih.gov/15777816/>, doi:10.1016/J.YHBEH.2005.01.002.
- [38] F. J. Sullo way. Darwin and his finches: The evolution of a legend. *Journal of the History of Biology*, 15(1):1–53, 3 1982. URL: <https://link.springer.com/article/10.1007/BF00132004>, doi:10.1007/BF00132004/METRICS.
- [39] R. W. Burkhardt. Niko Tinbergen: A Message in the Archives. *Journal of the history of biology*, 49(4):685–703, 12 2016. URL: <https://pubmed.ncbi.nlm.nih.gov/27435870/>, doi:10.1007/S10739-016-9450-Y.
- [40] A. M. Dettmer and A. J. Bennett. 100 Years of Comparative Psychology Advancing Practice, Policy, and the Public—and What the Future Requires. *Journal of Comparative Psychology*, 135(4):450–465, 2021. doi:10.1037/COM000299.
- [41] E. L. Thorndike. Animal intelligence: An experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, 2(4):i–109, 1898. doi:10.1037/H0092987.

- [42] B. F. Skinner. 'Superstition' in the pigeon. *Journal of Experimental Psychology*, 38(2):168–172, 4 1948. doi:10.1037/H0055873.
- [43] P. Georgiou, P. Zanos, T. C. M. Mou, X. An, D. M. Gerhard, D. I. Dryanovski, L. E. Potter, J. N. Highland, C. E. Jenne, B. W. Stewart, K. J. Pultorak, P. Yuan, C. F. Powels, J. Lovett, E. F. Pereira, S. M. Clark, L. H. Tonelli, R. Moaddel, C. A. Zarate, R. S. Duman, S. M. Thompson, and T. D. Gould. Experimenters' sex modulates mouse behaviors and neural responses to ketamine via corticotropin releasing factor. *Nature neuroscience*, 25(9):1191–1200, 9 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/36042309/>, doi:10.1038/S41593-022-01146-X.
- [44] E. J. Chesler, S. G. Wilson, W. R. Lariviere, S. L. Rodriguez-Zas, and J. S. Mogil. Influences of laboratory environment on behavior. *Nature neuroscience*, 5(11):1101–1102, 11 2002. URL: <https://pubmed.ncbi.nlm.nih.gov/12403996/>, doi:10.1038/NM1102-1101.
- [45] R. E. Sorge, L. J. Martin, K. A. Isbester, S. G. Sotocinal, S. Rosen, A. H. Tuttle, J. S. Wieskopf, E. L. Acland, A. Dokova, B. Kadoura, P. Leger, J. C. S. Mapplebeck, M. McPhail, A. Delaney, G. Wigerblad, A. P. Schumann, T. Quinn, J. Frasnelli, C. I. Svensson, W. F. Sternberg, and J. S. Mogil. Olfactory exposure to males, including men, causes stress and related analgesia in rodents. *Nature Methods* 2014 11:6, 11(6):629–632, 4 2014. URL: <https://www.nature.com/articles/nmeth.2935>, doi:10.1038/nmeth.2935.
- [46] E. M. Weber, J. A. Dallaire, B. N. Gaskill, K. R. Pritchett-Corning, and J. P. Garner. Aggression in group-housed laboratory mice: why can't we solve the problem? *Lab animal*, 46(4):157–161, 3 2017. URL: <https://pubmed.ncbi.nlm.nih.gov/28328884/>, doi:10.1038/LABAN.1219.
- [47] J. Bordes, L. Miranda, B. Müller-Myhsok, and M. V. Schmidt. Advancing social behavioral neuroscience by integrating ethology and comparative psychology methods through machine learning. *Neuroscience & Biobehavioral Reviews*, 151:105243, 8 2023. doi:10.1016/J.NEUBIOREV.2023.105243.
- [48] Y. Shemesh, Y. Sztainberg, O. Forkosh, T. Shlapobersky, A. Chen, and E. Schneidman. High-order social interactions in groups of mice. *eLife*, 2:759, 2013. doi:10.7554/eLife.00759.
- [49] O. Forkosh, S. Karamihalev, S. Roeh, U. Alon, S. Anpilov, C. Touma, M. Nussbaumer, C. Flachskamm, P. M. Kaplick, Y. Shemesh, and A. Chen. Identity domains capture individual differences from across the behavioral repertoire. *Nature Neuroscience*, 2019. URL: <https://doi.org/10.1038/s41593-019-0516-y>, doi:10.1038/s41593-019-0516-y.
- [50] D. A. Roberts, S. Yaida, and B. Hanin. The Principles of Deep Learning Theory. *The Principles of Deep Learning Theory*, 6 2021. URL: <http://>

BIBLIOGRAPHY

- arxiv.org/abs/2106.10165<http://dx.doi.org/10.1017/9781009023405>, doi: 10.1017/9781009023405.
- [51] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386–408, 11 1958. doi:10.1037/H0042519.
- [52] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [53] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature 1986 323:6088*, 323(6088):533–536, 1986. URL: <https://www.nature.com/articles/323533a0>, doi:10.1038/323533a0.
- [54] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25, 2012. URL: <http://code.google.com/p/cuda-convnet/>.
- [55] J. Chai, H. Zeng, A. Li, and E. W. Ngai. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6:100134, 12 2021. doi:10.1016/J.MLWA.2021.100134.
- [56] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, 6 2015. URL: <https://arxiv.org/abs/1506.01497v3>, doi:10.1109/TPAMI.2016.2577031.
- [57] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye. A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 14(8), 1 2020. URL: <https://arxiv.org/abs/2001.06937v1>, doi:10.1109/TKDE.2021.3130191.
- [58] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2022-June:10674–10685, 12 2021. URL: <https://arxiv.org/abs/2112.10752v2>, doi: 10.1109/CVPR52688.2022.01042.
- [59] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv*, 10 2020. URL: <https://arxiv.org/abs/2010.11929v2>.
- [60] M. Iman, K. Rasheed, and H. R. Arabnia. A Review of Deep Transfer Learning and Recent Advancements. *Technologies*, 11(2):40, 1 2022. URL: <http://arxiv.org/abs/2201.09679><http://dx.doi.org/10.3390/technologies11020040>, doi: 10.3390/technologies11020040.

- [61] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience* 2018 21:9, 21(9):1281–1289, 8 2018. URL: <https://www.nature.com/articles/s41593-018-0209-y>, doi:10.1038/s41593-018-0209-y.
- [62] T. D. Pereira, N. Tabris, A. Matsliah, D. M. Turner, J. Li, S. Ravindranath, E. S. Papadoyannis, E. Normand, D. S. Deutsch, Z. Y. Wang, G. C. McKenzie-Smith, C. C. Mitelut, M. D. Castro, J. D’Uva, M. Kislin, D. H. Sanes, S. D. Kocher, S. S. Wang, A. L. Falkner, J. W. Shaevitz, and M. Murthy. SLEAP: A deep learning system for multi-animal pose tracking. *Nature Methods* 2022 19:4, 19(4):486–495, 4 2022. URL: <https://www.nature.com/articles/s41592-022-01426-1>, doi:10.1038/s41592-022-01426-1.
- [63] M. Marks, Q. Jin, O. Sturman, L. von Ziegler, S. Kollmorgen, W. von der Behrens, V. Mante, J. Bohacek, and M. F. Yanik. Deep-learning-based identification, tracking, pose estimation and behaviour classification of interacting primates and mice in complex environments. *Nature Machine Intelligence* 2022 4:4, 4(4):331–340, 4 2022. URL: <https://www.nature.com/articles/s42256-022-00477-5>, doi:10.1038/s42256-022-00477-5.
- [64] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-December:770–778, 12 2015. URL: <https://arxiv.org/abs/1512.03385v1>, doi:10.1109/CVPR.2016.90.
- [65] D. Tuia, B. Kellenberger, S. Beery, B. R. Costelloe, S. Zuffi, B. Risse, A. Mathis, M. W. Mathis, F. van Langevelde, T. Burghardt, R. Kays, H. Klinck, M. Wikelski, I. D. Couzin, G. van Horn, M. C. Crofoot, C. V. Stewart, and T. Berger-Wolf. Perspectives in machine learning for wildlife conservation. *Nature Communications* 2022 13:1, 13(1):1–15, 2 2022. URL: <https://www.nature.com/articles/s41467-022-27980-y>, doi:10.1038/s41467-022-27980-y.
- [66] O. Sturman, L. von Ziegler, C. Schläppi, F. Akyol, M. Privitera, D. Slominski, C. Grimm, L. Thieren, V. Zerbi, B. Grewe, and J. Bohacek. Deep learning-based behavioral analysis reaches human accuracy and is capable of outperforming commercial solutions. *Neuropsychopharmacology*, 45(11):1942–1952, 7 2020. URL: <https://doi.org/10.3929/ethz-b-000431266>, doi:10.1038/S41386-020-0776-Y.
- [67] J. Lauer, M. Zhou, S. Ye, W. Menegas, S. Schneider, T. Nath, M. M. Rahman, V. Di Santo, D. Soberanes, G. Feng, V. N. Murthy, G. Lauder, C. Dulac, M. W. Mathis, and A. Mathis. Multi-animal pose estimation, identification and tracking with DeepLabCut. *Nature Methods* 2022 19:4, 19(4):496–504, 4 2022. URL: <https://www.nature.com/articles/s41592-022-01443-0>, doi:10.1038/s41592-022-01443-0.

BIBLIOGRAPHY

- [68] L. Miranda, J. Bordes, S. Gasperoni, and J. P. Lopez. Increasing resolution in stress neurobiology: from single cells to complex group behaviors. *Stress (Amsterdam, Netherlands)*, 26(1):2186141, 1 2023. URL: <https://pubmed.ncbi.nlm.nih.gov/36855966/>, doi:10.1080/10253890.2023.2186141.
- [69] S. Ye, A. Filippova, J. Lauer, M. Vidal, S. Schneider, T. Qiu, A. Mathis, . Mackenzie, and W. Mathis. SuperAnimal models pretrained for plug-and-play analysis of animal behavior. *arXiv*, 3 2022. URL: <https://arxiv.org/abs/2203.07436v2>.
- [70] F. de Chaumont, E. Ey, N. Torquet, T. Lagache, S. Dallongeville, A. Imbert, T. Legou, A. M. Le Sourd, P. Faure, T. Bourgeron, and J. C. Olivo-Marin. Real-time analysis of the behaviour of groups of mice via a depth-sensing camera and machine learning. *Nature biomedical engineering*, 3(11):930–942, 11 2019. URL: <https://pubmed.ncbi.nlm.nih.gov/31110290/>, doi:10.1038/S41551-019-0396-1.
- [71] S. Ro, N. †1, N. L. Goodwin, J. J. Choong, S. Hwang, H. R. Wright, Z. C. Norville, X. Tong, D. Lin, B. S. Bentzley, N. Eshel, R. J. McLaughlin, and S. A. Golden. Simple Behavioral Analysis (SimBA) – an open source toolkit for computer classification of complex social behaviors in experimental animals. *bioRxiv*, 2020. URL: <https://doi.org/10.1101/2020.04.19.049452>, doi:10.1101/2020.04.19.049452.
- [72] D. M. Burns and C. M. Whyne. Seglearn: A Python Package for Learning Sequences and Time Series. *Journal of Machine Learning Research*, 19, 3 2018. URL: <https://arxiv.org/abs/1803.08118v3>.
- [73] C. Segalin, J. Williams, T. Karigo, M. Hui, M. Zelikowsky, J. J. Sun, P. Perona, D. J. Anderson, and A. Kennedy. The mouse action recognition system (MARS) software pipeline for automated analysis of social behaviors in mice. *eLife*, 10, 11 2021. doi:10.7554/ELIFE.63720.
- [74] T. R. Shultz and S. E. Fahlman. Curse of Dimensionality. *Encyclopedia of Machine Learning and Data Mining*, pages 314–315, 2017. URL: https://link.springer.com/referenceworkentry/10.1007/978-1-4899-7687-1_192, doi:10.1007/978-1-4899-7687-1{_}192.
- [75] J. Bordes, L. Miranda, M. Reinhardt, S. Narayan, J. Hartmann, E. L. Newman, L. M. Brix, L. van Doeselaar, C. Engelhardt, L. Dillmann, S. Mitra, K. J. Ressler, B. Pütz, F. Agakov, B. Müller-Myhsok, and M. V. Schmidt. Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress. *Nature Communications 2023 14:1*, 14(1):1–19, 7 2023. URL: <https://www.nature.com/articles/s41467-023-40040-3>, doi:10.1038/s41467-023-40040-3.
- [76] C. Weinreb, M. Abdal, M. Osman, L. Zhang, S. Lin, J. Pearl, S. Annapragada, E. Conlin, W. F. Gillis, M. Jay, S. Ye, A. Mathis, M. W. Mathis, T. Pereira, S. W. Linderman, and S. R. Datta. Keypoint-MoSeq: parsing behavior by linking point

- tracking to pose dynamics. *arXiv*, 2023. URL: <https://doi.org/10.1101/2023.03.16.532307>, doi:10.1101/2023.03.16.532307.
- [77] J. F. Schweihoff, A. I. Hsu, M. K. Schwarz, and E. A. Yttri. A-SOiD, an active learning platform for expert-guided, data efficient discovery of behavior. *bioRxiv*, page 2022.11.04.515138, 11 2022. URL: <https://www.biorxiv.org/content/10.1101/2022.11.04.515138v2><https://www.biorxiv.org/content/10.1101/2022.11.04.515138v2.abstract>, doi:10.1101/2022.11.04.515138.
- [78] A. I. Hsu and E. A. Yttri. B-SOiD, an open-source unsupervised algorithm for identification and fast prediction of behaviors. *Nature Communications*, 12(1), 12 2021. URL: [/pmc/articles/PMC8408193/](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8408193/)[https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8408193/](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8408193/?report=abstract), doi:10.1038/S41467-021-25420-X.
- [79] A. B. Wiltschko, T. Tsukahara, A. Zeine, R. Anyoha, W. F. Gillis, J. E. Markowitz, R. E. Peterson, J. Katon, M. J. Johnson, and S. R. Datta. Revealing the structure of pharmacobehavioral space through motion sequencing. *Nature Neuroscience*, 2020. URL: <https://doi.org/10.1038/s41593-020-00706-3>, doi:10.1038/s41593-020-00706-3.
- [80] K. Luxem, P. Mocellin, F. Fuhrmann, J. Kürsch, S. R. Miller, J. J. Palop, S. Remy, and P. Bauer. Identifying behavioral structure from deep variational embeddings of animal motion. *Communications Biology* 2022 5:1, 5(1):1–15, 11 2022. URL: <https://www.nature.com/articles/s42003-022-04080-7>, doi:10.1038/s42003-022-04080-7.
- [81] L. Rubio, E. Palomo, E. Domínguez, S. Chen, and W. Guo. Auto-Encoders in Deep Learning—A Review with New Perspectives. *Mathematics* 2023, Vol. 11, Page 1777, 11(8):1777, 4 2023. URL: <https://www.mdpi.com/2227-7390/11/8/1777/html><https://www.mdpi.com/2227-7390/11/8/1777>, doi:10.3390/MATH11081777.
- [82] N. L. Goodwin, S. R. Nilsson, J. J. Choong, and S. A. Golden. Toward the explainability, transparency, and universality of machine learning for behavioral classification in neuroscience. *Current opinion in neurobiology*, 73, 4 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/35487088/>, doi:10.1016/J.CONB.2022.102544.
- [83] G. Sanacora, Z. Yan, and M. Popoli. The stressed synapse 2.0: pathophysiological mechanisms in stress-related neuropsychiatric disorders. *Nature Reviews Neuroscience* 2021 23:2, 23(2):86–103, 12 2021. URL: <https://www.nature.com/articles/s41583-021-00540-x>, doi:10.1038/s41583-021-00540-x.
- [84] L. Musazzi and J. Marrocco. The Many Faces of Stress: Implications for Neuropsychiatric Disorders. *Neural Plasticity*, 2016, 2016. URL: <https://www.hindawi.com/journals/np/2016/8389737/>, doi:10.1155/2016/8389737.

BIBLIOGRAPHY

- [85] M. T. Davis, S. E. Holmes, R. H. Pietrzak, and I. Esterlis. Neurobiology of Chronic Stress-Related Psychiatric Disorders: Evidence from Molecular Imaging Studies. <http://dx.doi.org/10.1177/2470547017710916>, 1, 6 2017. URL: <https://journals.sagepub.com/doi/10.1177/2470547017710916>, doi:10.1177/2470547017710916.
- [86] L. Musazzi, P. Tornese, N. Sala, and M. Popoli. What acute stress protocols can tell us about PTSD and stress-related neuropsychiatric disorders. *Frontiers in Pharmacology*, 9(JUN):758, 7 2018. doi:10.3389/FPHAR.2018.00758/BIBTEX.
- [87] L. D. Godoy, M. T. Rossignoli, P. Delfino-Pereira, N. Garcia-Cairasco, and E. H. d. L. Umeoka. A comprehensive overview on stress neurobiology: Basic concepts and clinical implications. *Frontiers in Behavioral Neuroscience*, 12:127, 7 2018. doi:10.3389/FNBEH.2018.00127/BIBTEX.
- [88] W. B. Cannon. Bodily changes in pain, hunger, fear and rage. An account of recent researches into the function of emotional excitement. *D. Appleton & Company*, 1953.
- [89] World Health Organization. Depression and Other Common Mental Disorders Global Health Estimates. Technical report, World Health Organization, 2021.
- [90] Y. Shemesh and A. Chen. A paradigm shift in translational psychiatry through rodent neuroethology. *Molecular Psychiatry* 2023 28:3, 28(3):993–1003, 1 2023. URL: <https://www.nature.com/articles/s41380-022-01913-z>, doi:10.1038/s41380-022-01913-z.
- [91] M. A. Crocq. A history of anxiety: from Hippocrates to DSM. *Dialogues in Clinical Neuroscience*, 17(3):319, 2015. URL: </pmc/articles/PMC4610616/>[https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4610616/](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4610616/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC4610616/), doi:10.31887/DCNS.2015.17.3/MACROCQ.
- [92] J. P. Lopez, E. Brivio, A. Santambrogio, C. De Donno, A. Kos, M. Peters, N. Rost, D. Czamara, T. M. Brückl, S. Roeh, M. L. Pöhlmann, C. Engelhardt, A. Ressler, R. Stoffel, A. Tontsch, J. M. Villamizar, M. Reincke, A. Riester, S. Sbiera, M. Fassnacht, H. S. Mayberg, W. E. Craighead, B. W. Dunlop, C. B. Nemeroff, M. V. Schmidt, E. B. Binder, F. J. Theis, F. Beuschlein, C. L. Andoniadou, and A. Chen. Single-cell molecular profiling of all three components of the HPA axis reveals adrenal ABCB1 as a regulator of stress adaptation. *Science Advances*, 7(5):4497–4524, 1 2021. URL: <https://www.science.org/doi/10.1126/sciadv.abe4497>, doi:10.1126/SCIADV.ABE4497/SUPPL{_}FILE/ABE4497{_}SM.PDF.
- [93] X. Li, L. Peng, P. Xi, X. Hua, H. Su, S. Wangcheng, C. Yu, H. Wu, H. Li, Y. Ren, X. Chen, L. Liang, Z. Zhang, R. Chen Wuhan Jinyintan Hospital, C. Fei Deng, G. Qu, R. Wang, Y. Wang, X. Zhou Hubei, F. Wang, J. Zhao, a. N. Engl J Med, X.-J. Wang, H. Hu, C. Huang, H. Kennedy, C. Tony Li, N. Logothetis, Z.-L. Lu, Q. Luo, M.-m. Poo, D. Tsao, S. Wu, Z. Wu, X. Zhang, and D. Zhou. Computational

- neuroscience: a frontier of the 21st century. *National Science Review*, 7(9):1418–1422, 9 2020. URL: <https://academic.oup.com/nsr/article/7/9/1418/5856589>, doi:10.1093/NSR/NWAA129.
- [94] F. P. Binder and V. I. Spoormaker. Quantifying Human Avoidance Behavior in Immersive Virtual Reality. *Frontiers in Behavioral Neuroscience*, 14:163, 9 2020. doi:10.3389/FNBEH.2020.569899/BIBTEX.
- [95] F. P. Binder, D. Pöhlchen, P. Zwanzger, and V. I. Spoormaker. Facing Your Fear in Immersive Virtual Reality: Avoidance Behavior in Specific Phobia. *Frontiers in Behavioral Neuroscience*, 16:91, 4 2022. doi:10.3389/FNBEH.2022.827673/BIBTEX.
- [96] S. J. Russo and E. J. Nestler. The brain reward circuitry in mood disorders. *Nature reviews. Neuroscience*, 14(9):609–625, 9 2013. URL: <https://pubmed.ncbi.nlm.nih.gov/23942470/>, doi:10.1038/NRN3381.
- [97] S. A. Golden, H. E. Covington, O. Berton, and S. J. Russo. A standardized protocol for repeated social defeat stress in mice. *Nature protocols*, 6(8):1183–1191, 7 2011. URL: <https://pubmed.ncbi.nlm.nih.gov/21799487/>, doi:10.1038/NPROT.2011.361.
- [98] N. N. Kudryavtseva, I. V. Bakshtanovskaya, and L. A. Koryakina. Social model of depression in mice of C57BL/6J strain. *Pharmacology, biochemistry, and behavior*, 38(2):315–320, 1991. URL: <https://pubmed.ncbi.nlm.nih.gov/2057501/>, doi:10.1016/0091-3057(91)90284-9.
- [99] S. D. Iñiguez, L. M. Riggs, S. J. Nieto, G. Dayrit, N. N. Zamora, K. L. Shawhan, B. Cruz, and B. L. Warren. Social defeat stress induces a depression-like phenotype in adolescent male c57BL/6 mice. *Stress (Amsterdam, Netherlands)*, 17(3):247–255, 2014. URL: <https://pubmed.ncbi.nlm.nih.gov/24689732/>, doi:10.3109/10253890.2014.910650.
- [100] K. Yoshida, M. R. Drew, A. Kono, M. Mimura, N. Takata, and K. F. Tanaka. Chronic social defeat stress impairs goal-directed behavior through dysregulation of ventral hippocampal activity in male mice. *Neuropsychopharmacology : official publication of the American College of Neuropsychopharmacology*, 46(9):1606–1616, 8 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/33692477/>, doi:10.1038/S41386-021-00990-Y.
- [101] R. J. Donahue, J. W. Muschamp, S. J. Russo, E. J. Nestler, and W. A. Carlezon. Effects of striatal Δ FosB overexpression and ketamine on social defeat stress-induced anhedonia in mice. *Biological psychiatry*, 76(7):550–558, 2014. URL: <https://pubmed.ncbi.nlm.nih.gov/24495460/>, doi:10.1016/J.BIOPSYCH.2013.12.014.
- [102] J. P. Lopez, M. D. Lücken, E. Brivio, S. Karamihalev, A. Kos, C. De Donno, A. Benjamin, H. Yang, A. L. Dick, R. Stoffel, C. Flachskamm, A. Ressler, S. Roeh, R. E. Huettl, A. Parl, C. Eggert, B. Novak, Y. Yan, K. Yeoh, M. Holzapfel, B. Hauger,

BIBLIOGRAPHY

- D. Harbich, B. Schmid, R. Di Giaimo, C. W. Turck, M. V. Schmidt, J. M. Deussing, M. Eder, J. Dine, F. J. Theis, and A. Chen. Ketamine exerts its sustained antidepressant effects via cell-type-specific regulation of Kcnq2. *Neuron*, 110(14):2283–2298, 7 2022. URL: <http://www.cell.com/article/S0896627322004093/fulltext><http://www.cell.com/article/S0896627322004093/abstract>[https://www.cell.com/neuron/abstract/S0896-6273\(22\)00409-3](https://www.cell.com/neuron/abstract/S0896-6273(22)00409-3), doi:10.1016/j.neuron.2022.05.001.
- [103] B. Lafabregue, J. Weber, P. Gañarski, G. Forestier, B. Lafabregue, J. Weber, G. Forestier, P. Gañarski, A. Hoang, A. Dau, K. Bagnall, C.-C. M. Kamgar, Y. Yeh, S. Zhu, A. Gharghabi, E. Ratanamahatana, and M. B. Keogh. End-to-end deep representation learning for time series clustering: a comparative study. *Data Mining and Knowledge Discovery*, 2022. URL: <https://doi.org/10.1007/s10618-021-00796-y>, doi:10.1007/s10618-021-00796-y.
- [104] K. P. Murphy. *Probabilistic Machine Learning: an introduction*. MIT Press, 2022. URL: <https://lccn.loc.gov/2021027430>.
- [105] M. Christ, N. Braun, J. Neuffer, and A. W. Kempa-Liehr. Time Series Feature Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing*, 307:72–77, 9 2018. doi:10.1016/J.NEUCOM.2018.03.067.
- [106] R. Tavenard. An introduction to Dynamic Time Warping, 2021. URL: <https://rtavenar.github.io/blog/dtw.html>.
- [107] B. D. Fulcher. Feature-based time-series analysis. *arXiv*, 2017.
- [108] J. Xia, Y. Zhang, J. Song, Y. Chen, Y. Wang, and S. Liu. Revisiting Dimensionality Reduction Techniques for Visual Cluster Analysis: An Empirical Study. *arXiv*, 2021. URL: <https://github.com/DR-approach/DR-approaches>.
- [109] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv*, 2 2018. URL: <https://arxiv.org/abs/1802.03426v3>.
- [110] J. Enes, R. R. Expósito, J. Fuentes, J. L. Cacheiro, and J. Touriño. A pipeline architecture for feature-based unsupervised clustering using multivariate time series from HPC jobs. *Information Fusion*, 93:1–20, 5 2023. doi:10.1016/J.INFFUS.2022.12.017.
- [111] G. Petelin, G. Cenikj, and T. Eftimov. Towards understanding the importance of time-series features in automated algorithm performance prediction. *Expert Systems with Applications*, 213:119023, 3 2023. doi:10.1016/J.ESWA.2022.119023.
- [112] T. K. Vintsyuk. Speech discrimination by dynamic programming. *Cybernetics*, 4(1):52–57, 1 1968. URL: <https://link.springer.com/article/10.1007/BF01074755>, doi:10.1007/BF01074755/METRICS.

- [113] F. Itakura. Minimum Prediction Residual Principle Applied to Speech Recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):67–72, 1975. doi:10.1109/TASSP.1975.1162641.
- [114] H. Sakoe and S. Chiba. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978. doi:10.1109/TASSP.1978.1163055.
- [115] F. Petitjean, A. Ketterlin, and P. Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3):678–693, 3 2011. doi:10.1016/J.PATCOG.2010.09.013.
- [116] C. J. Tralie and E. Dempsey. Exact, Parallelizable Dynamic Time Warping Alignment with Linear Memory. *arXiv*, 8 2020. URL: <https://arxiv.org/abs/2008.02734v1>.
- [117] K. Murphy. *Probabilistic Machine Learning: advanced topics*. MIT press, 2023. URL: <https://mitpress.mit.edu/search-result-list/>.
- [118] B. Mor, S. Garhwal, and A. Kumar. A Systematic Review of Hidden Markov Models and Their Applications. *Archives of Computational Methods in Engineering*, 28(3):1429–1448, 5 2021. URL: <https://link.springer.com/article/10.1007/s11831-020-09422-4>, doi:10.1007/S11831-020-09422-4/TABLES/5.
- [119] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, 12 2013. URL: <https://arxiv.org/abs/1312.6114v11>.
- [120] P. H. Le-Khac, G. Healy, and A. F. Smeaton. Contrastive Representation Learning: A Framework and Review. *IEEE Access*, 8:193907–193934, 10 2020. URL: <http://arxiv.org/abs/2010.05113><http://dx.doi.org/10.1109/ACCESS.2020.3031549>, doi:10.1109/ACCESS.2020.3031549.
- [121] Z. C. Lipton, J. Berkowitz, and C. Elkan. A Critical Review of Recurrent Neural Networks for Sequence Learning. *arXiv*, 5 2015. URL: <https://arxiv.org/abs/1506.00019v4>.
- [122] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1724–1734, 6 2014. URL: <https://arxiv.org/abs/1406.1078v3>, doi:10.3115/v1/d14-1179.
- [123] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 11 1997. URL: <https://direct.mit.edu/neco/article/9/8/1735/6109/Long-Short-Term-Memory>, doi:10.1162/NECO.1997.9.8.1735.

BIBLIOGRAPHY

- [124] S. Bai, J. Z. Kolter, and V. Koltun. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv*, 3 2018. URL: <https://arxiv.org/abs/1803.01271v2>.
- [125] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. WaveNet: A Generative Model for Raw Audio. *arXiv*, 9 2016. URL: <https://arxiv.org/abs/1609.03499v2>.
- [126] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 2017-December:5999–6009, 6 2017. URL: <https://arxiv.org/abs/1706.03762v5>.
- [127] L. McInnes, J. Healy, and S. Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 3 2017. URL: <https://joss.theoj.org/papers/10.21105/joss.00205>, doi:10.21105/JOSS.00205.
- [128] S. R. Datta. Q&A: Understanding the composition of behavior. *BMC biology*, 2015. URL: <https://doi.org/10.1186/s12915-019-0663-3>, doi:10.1186/s12915-019-0663-3.
- [129] D. R. Maciver and Z. Hatfield-Dodds. Hypothesis: A new approach to property-based testing. *Journal of Open Source Software*, 2019. URL: <https://pypi.org/project/hypothesis-bio/>, doi:10.21105/joss.01891.
- [130] Coverage.py — Coverage.py 7.2.5 documentation. URL: <https://coverage.readthedocs.io/en/7.2.5/>.
- [131] Home — Read the Docs. URL: <https://readthedocs.org/>.
- [132] X. Jiang, R. Zhu, P. Ji, S. Li, and S. Member. Co-embedding of Nodes and Edges with Graph Neural Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, X:1–1, 10 2020. URL: <https://arxiv.org/abs/2010.13242v1>, doi:10.1109/tpami.2020.3029762.
- [133] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian. Multiscale Spatio-Temporal Graph Neural Networks for 3D Skeleton-Based Motion Prediction. *IEEE Transactions on Image Processing*, 30:7760–7775, 8 2021. URL: <https://arxiv.org/abs/2108.11244v1>, doi:10.1109/tip.2021.3108708.
- [134] Z. A. Sahili and M. Awad. Spatio-Temporal Graph Neural Networks: A Survey. *arxiv*, 1 2023. URL: <https://arxiv.org/abs/2301.10569v2>.
- [135] Z. Jiang, Y. Zheng, H. Tan, B. Tang, and H. Zhou. Variational Deep Embedding: An Unsupervised and Generative Approach to Clustering. *IJCAI International Joint Conference on Artificial Intelligence*, 0:1965–1972, 11 2016. URL: <https://arxiv.org/abs/1611.05148v3>, doi:10.24963/ijcai.2017/273.

- [136] L. Manduchi, R. Marcinkevičs, and J. E. Vogt. A deep variational approach to clustering survival data. *ICLR*, 2021.
- [137] A. Van Den Oord, O. Vinyals, and K. Kavukcuoglu. Neural Discrete Representation Learning. *Advances in Neural Information Processing Systems*, 2017-December:6307–6316, 11 2017. URL: <https://arxiv.org/abs/1711.00937v2>.
- [138] V. Fortuin, M. Hüser, F. Locatello, H. Strathmann, and G. Rätsch. SOM-VAE: Interpretable Discrete Representation Learning on Time Series. *7th International Conference on Learning Representations, ICLR 2019*, 6 2018. URL: <https://arxiv.org/abs/1806.02199v7>.
- [139] J. C. Bezdek, R. Ehrlich, and W. Full. FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203, 1 1984. doi:10.1016/0098-3004(84)90020-7.
- [140] S. Deldari, D. V. Smith, H. Xue, and F. D. Salim. Time series change point detection with self-supervised contrastive predictive coding. *The Web Conference 2021 - Proceedings of the World Wide Web Conference, WWW 2021*, pages 3124–3135, 4 2021. URL: <https://dl.acm.org/doi/10.1145/3442381.3449903>, doi:10.1145/3442381.3449903.
- [141] A. van den Oord DeepMind, Y. Li DeepMind, and O. Vinyals DeepMind. Representation Learning with Contrastive Predictive Coding. *arxiv*, 2019.
- [142] J. Schreiber and P. G. Allen. Pomegranate: fast and flexible probabilistic modeling in python. *Journal of Machine Learning Research*, 18:1–6, 10 2017. URL: <https://arxiv.org/abs/1711.00137v2>.
- [143] L. Miranda, J. Bordes, B. Pütz, M. V. Schmidt, and B. Müller-Myhsok. DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data. *Journal of Open Source Software*, 8(86):5394, 6 2023. URL: <https://joss.theoj.org/papers/10.21105/joss.05394>, doi:10.21105/JOSS.05394.
- [144] G. Ceballos, P. R. Ehrlich, and P. H. Raven. Vertebrates on the brink as indicators of biological annihilation and the sixth mass extinction. *Proceedings of the National Academy of Sciences of the United States of America*, 117(24):13596–13602, 6 2020. URL: <https://pmc/articles/PMC7306750/>[https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7306750/](https://pmc/articles/PMC7306750/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC7306750/), doi:10.1073/PNAS.1922686117/-/DCSUPPLEMENTAL.
- [145] Z. Shao, R. Zhao, S. Yuan, M. Ding, and Y. Wang. Tracing the evolution of AI in the past decade and forecasting the emerging trends. *Expert Systems with Applications*, 209:118221, 12 2022. doi:10.1016/J.ESWA.2022.118221.
- [146] J. J. S. Caltech, T. Karigo, D. Chakraborty, A. Research, S. P. Mohanty, B. Wild, Q. Sun, C. Chen, D. J. Anderson, P. P. Caltech, Y. Yue, and A. Kennedy. The

BIBLIOGRAPHY

- Multi-Agent Behavior Dataset: Mouse Dyadic Social Interactions. *arxiv*, 4 2021. URL: <https://arxiv.org/abs/2104.02710v4>.
- [147] ANY-maze. URL: <https://www.any-maze.com/>.
- [148] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, A. Zisserman, and G. Deepmind. TAPIR: Tracking Any Point with per-frame Initialization and temporal Refinement. URL: <https://www.github.com/>.
- [149] Q. Wang, Y.-Y. Chang, R. Cai, Z. Li, B. Hariharan, A. Holynski, and N. Snavely. Tracking Everything Everywhere All at Once.
- [150] L. Weidinger, J. Mellor, M. Rauh, C. Griffin, J. Uesato, P.-S. Huang, M. Cheng, M. Glaese, B. Balle, A. Kasirzadeh, Z. Kenton, S. Brown, W. Hawkins, T. Stepleton, C. Biles, A. Birhane, J. Haas, L. Rimell, L. A. Hendricks, W. Isaac, S. Legassick, G. Irving, I. Gabriel, and I. Com . Ethical and social risks of harm from Language Models. *arxiv*, 12 2021. URL: <https://arxiv.org/abs/2112.04359v1>.
- [151] L. Weidinger, J. Uesato, M. Rauh, C. Griffin, J. Mellor, A. Glaese, M. Cheng, B. Balle, A. Kasirzadeh, C. Biles, S. U. Brown DeepMind Zac Kenton DeepMind, W. Hawkins, T. Stepleton, A. Birhane, L. Anne Hendricks, L. Rimell, W. Isaac, J. Haas, S. Legassick, G. Irving, I. Gabriel, P.-S. Huang, S. Brown, and Z. Kenton. Taxonomy of Risks posed by Language Models; Taxonomy of Risks posed by Language Models. *FAccT*, 22, 2022. URL: <https://doi.org/10.1145/3531146.3533088>, doi:10.1145/3531146.3533088.
- [152] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. Chatterji, A. Chen, K. Creel, J. Q. Davis, D. Demszky, C. Donahue, M. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. Nyarko, G. Ogut, L. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. R , D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tram r, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang. On the Opportunities and Risks of Foundation Models. *arxiv*, 8 2021. URL: <https://arxiv.org/abs/2108.07258v3>.
- [153] OpenAI. GPT-4 Technical Report. *arxiv*, 3 2023. URL: <https://arxiv.org/abs/2303.08774v3>.

- [154] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arxiv*, 4 2022. URL: <https://arxiv.org/abs/2204.06125v1>.
- [155] F. A. Wolf, P. Angerer, and F. J. Theis. SCANPY: Large-scale single-cell gene expression data analysis. *Genome Biology*, 19(1):1–5, 2 2018. URL: <https://genomebiology.biomedcentral.com/articles/10.1186/s13059-017-1382-0>, doi:10.1186/S13059-017-1382-0/FIGURES/1.
- [156] Y. Hao, S. Hao, E. Andersen-Nissen, W. M. Mauck, S. Zheng, A. Butler, M. J. Lee, A. J. Wilk, C. Darby, M. Zager, P. Hoffman, M. Stoeckius, E. Papalexi, E. P. Mimitou, J. Jain, A. Srivastava, T. Stuart, L. M. Fleming, B. Yeung, A. J. Rogers, J. M. McElrath, C. A. Blish, R. Gottardo, P. Smibert, and R. Satija. Integrated analysis of multimodal single-cell data. *Cell*, 184(13):3573–3587, 6 2021. doi:10.1016/J.CELL.2021.04.048.
- [157] G. Palla, H. Spitzer, M. Klein, D. Fischer, A. C. Schaar, L. B. Kuemmerle, S. Rybakov, I. L. Ibarra, O. Holmberg, I. Virshup, M. Lotfollahi, S. Richter, and F. J. Theis. Squidpy: a scalable framework for spatial omics analysis. *Nature methods*, 19(2):171–178, 2 2022. URL: <https://pubmed.ncbi.nlm.nih.gov/35102346/>, doi:10.1038/S41592-021-01358-2.
- [158] M. Mann, C. Kumar, W. F. Zeng, and M. T. Strauss. Artificial intelligence for proteomics and biomarker discovery. *Cell systems*, 12(8):759–770, 8 2021. URL: <https://pubmed.ncbi.nlm.nih.gov/34411543/>, doi:10.1016/J.CELS.2021.06.006.
- [159] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature 2021 596:7873*, 596(7873):583–589, 7 2021. URL: <https://www.nature.com/articles/s41586-021-03819-2>, doi:10.1038/s41586-021-03819-2.
- [160] P. T. Toi, H. J. Jang, K. Min, S. P. Kim, S. K. Lee, J. Lee, J. Kwag, and J. Y. Park. In vivo direct imaging of neuronal activity at high temporospatial resolution. *Science*, 378(6616), 10 2022. URL: <https://www.science.org/doi/10.1126/science.abh4340>, doi:10.1126/SCIENCE.ABH4340/SUPPL{_}FILE/SCIENCE.ABH4340{_}MDAR{_}REPRODUCIBILITY{_}CHECKLIST.PDF.
- [161] H. Dana, Y. Sun, B. Mohar, B. K. Hulse, A. M. Kerlin, J. P. Hasseman, G. Tsegaye, A. Tsang, A. Wong, R. Patel, J. J. Macklin, Y. Chen, A. Konnerth, V. Jayaraman, L. L. Looger, E. R. Schreier, K. Svoboda, and D. S. Kim. High-performance calcium sensors for imaging activity in neuronal populations and microcompartments.

BIBLIOGRAPHY

- Nature Methods* 2019 16:7, 16(7):649–657, 6 2019. URL: <https://www.nature.com/articles/s41592-019-0435-6>, doi:10.1038/s41592-019-0435-6.
- [162] A. de Groot, B. J. van den Boom, R. M. van Genderen, J. Coppens, J. van Veldhuijzen, J. Bos, H. Hoedemaker, M. Negrello, I. Willuhn, C. I. De Zeeuw, and T. M. Hoogland. Ninscope, a versatile miniscope for multi-region circuit investigations. *eLife*, 9, 1 2020. doi:10.7554/ELIFE.49987.
- [163] S. Jabeen, X. Li, M. S. Amin, O. Bourahla, S. Li, and A. Jabbar. A Review on Methods and Applications in Multimodal Deep Learning. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 19(2s):1–41, 2 2022. URL: <https://arxiv.org/abs/2202.09195v1>, doi:10.1145/3545572.
- [164] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan. Ensemble deep learning: A review. *arxiv*, 2022.
- [165] S. Schneider, J. H. Lee, and M. W. Mathis. Learnable latent embeddings for joint behavioural and neural analysis. *Nature* 2023 617:7960, 617(7960):360–368, 5 2023. URL: <https://www.nature.com/articles/s41586-023-06031-6>, doi:10.1038/s41586-023-06031-6.
- [166] B. Chu, K. Marwaha, T. Sanvictores, and D. Ayers. Physiology, Stress Reaction. *StatPearls*, 9 2022. URL: <https://www.ncbi.nlm.nih.gov/books/NBK541120/>.
- [167] O. Abiola, J. M. Angel, P. Avner, A. A. Bachmanov, J. K. Belknap, B. Bennett, E. P. Blankenhorn, D. A. Blizard, V. Bolivar, G. A. Brockmann, K. J. Buck, J. F. Bureau, W. L. Casley, E. J. Chesler, J. M. Cheverud, G. A. Churchill, M. Cook, J. C. Crabbe, W. E. Crusio, A. Darvasi, G. de Haan, P. Demant, R. W. Doerge, R. W. Elliott, C. R. Farber, L. Flaherty, J. Flint, H. Gershenfeld, J. P. Gibson, J. Gu, W. Gu, H. Himmelbauer, R. Hitzemann, H. C. Hsu, K. Hunter, F. A. Iraqi, R. C. Jansen, T. E. Johnson, B. C. Jones, G. Kempermann, F. Lammert, L. Lu, K. F. Manly, D. B. Matthews, J. F. Medrano, M. Mehrabian, G. Mittleman, B. A. Mock, J. S. Mogil, X. Montagutelli, G. Morahan, J. D. Mountz, H. Nagase, R. S. Nowakowski, B. F. O’Hara, A. V. Osadchuk, B. Paigen, A. A. Palmer, J. L. Peirce, D. Pomp, M. Rosemann, G. D. Rosen, L. C. Schalkwyk, Z. Seltzer, S. Settle, K. Shimomura, S. Shou, J. M. Sikela, L. D. Siracusa, J. L. Spearow, C. Teuscher, D. W. Threadgill, L. A. Toth, A. A. Toye, C. Vadasz, G. Van Zant, E. Wakeland, R. W. Williams, H. G. Zhang, and F. Zou. The nature and identification of quantitative trait loci: a community’s view. *Nature reviews. Genetics*, 4(11):911, 11 2003. URL: <https://pmc/articles/PMC2063446/>, doi:10.1038/NGR1206.
- [168] Industrializing behavioral testing to redefine neuropsychiatric drug discovery. URL: <https://www.nature.com/articles/d43747-022-00182-7>.

List of published PhD papers

- [PhD1] L. Miranda, R. Paul, B. Pütz, N. Koutsouleris, and B. Müller-Myhsok. Systematic review of functional MRI applications for psychiatric disease subtyping. *Frontiers in Psychiatry*, 12, October 2021. URL: <https://doi.org/10.3389/fpsy.2021.665536>, doi:10.3389/fpsy.2021.665536.
- [PhD2] L. Miranda, J. Bordes, S. Gasperoni, and J. P. Lopez. Increasing resolution in stress neurobiology: from single cells to complex group behaviors. *Stress*, 26(1), January 2023. URL: <https://doi.org/10.1080/10253890.2023.2186141>, doi:10.1080/10253890.2023.2186141.
- [PhD3] J. Bordes, L. Miranda, B. Müller-Myhsok, and M. V. Schmidt. Advancing social behavioral neuroscience by integrating ethology and comparative psychology methods through machine learning. *Neuroscience & Biobehavioral Reviews*, 151:105243, 8 2023. doi:10.1016/J.NEUBIOREV.2023.105243.
- [PhD4] L. Miranda, J. Bordes, B. Pütz, M. V. Schmidt, and B. Müller-Myhsok. DeepOF: a Python package for supervised and unsupervised pattern recognition in mice motion tracking data. *Journal of Open Source Software*, 8(86):5394, 6 2023. URL: <https://joss.theoj.org/papers/10.21105/joss.05394>, doi:10.21105/JOSS.05394.
- [PhD5] J. Bordes, L. Miranda, M. Reinhardt, S. Narayan, J. Hartmann, E. L. Newman, L. M. Brix, L. van Doeselaar, C. Engelhardt, L. Dillmann, S. Mitra, K. J. Ressler, B. Pütz, F. Agakov, B. Müller-Myhsok, and M. V. Schmidt. Automatically annotated motion tracking identifies a distinct social behavioral profile following chronic social defeat stress. *Nature Communications*, 14(1), July 2023. URL: <https://doi.org/10.1038/s41467-023-40040-3>, doi:10.1038/s41467-023-40040-3.