Distribution Agreement

In presenting this thesis or dissertation as a partial fulfillment of the requirements for an advanced degree from Emory University, I hereby grant to Emory University and its agents the non-exclusive license to archive, make accessible, and display my thesis or dissertation in whole or in part in all forms of media, now or hereafter known, including display on the world wide web. I understand that I may select some access restrictions as part of the online submission of this thesis or dissertation. I retain all ownership rights to the copyright of the thesis or dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

Signature:

_____ _____

Josuan Calderon Date

Dynamical Inference and Representations in Complex Biological Systems

By

Josuan Calderon
Doctor of Philosophy

Physics

Gordon J. Berman, Ph.D.
Advisor

Ilya Nemenman, Ph.D.
Committee Member

Justin C. Burton, Ph.D.
Committee Member

Shella D. Keilholz, Ph.D.
Committee Member

Daniel M. Sussman, Ph.D.
Committee Member

Accepted:

Kimberly J. Arriola, Ph.D.
Dean of the James T. Laney School of Graduate Studies

Date

Dynamical Inference and Representations in Complex Biological Systems

By

Josuan Calderon
B.S. in Physics, Florida International University, FL, 2016
B.S. in Mathematics, Florida International University, FL, 2016

Advisor: Gordon J. Berman, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Physics
2023

Abstract

Dynamical Inference and Representations in Complex Biological Systems
By Josuan Calderon

This thesis introduces a comprehensive analytical framework that combines Temporal Autoencoders for Causal Inference (TACI) with a new pipeline for discovering and analyzing behavioral states in complex dynamical systems. Although these two methodologies have different purposes, they both focus on comprehending time-dependent phenomena through the lens of time series and temporal interactions, which are omnipresent in the natural world.

We begin with TACI, a novel methodology designed to analyze time-varying causal interactions within complex dynamical systems. Traditional approaches to causality often fall short when faced with non-linear, non-stationary interactions between system variables. To address these challenges, TACI leverages a novel metric, the Comparative Surrogate Granger Index (CGSI), alongside a two-headed Temporal Convolutional Network (TCN) autoencoder architecture. Through tests on both synthetic and real-world datasets, we demonstrate TACI's ability to accurately quantify dynamic causal interactions across a variety of systems. Our findings display the method's effectiveness compared to existing approaches and also enhance our understanding of time-varying interactions in various domains, from physical to biological systems. Through this work, TACI emerges as a significant advancement in the field of causal inference, promising to deepen our comprehension of dynamic systems across a range of scientific disciplines.

The thesis also explores the intricate dynamics of behavioral states using a comprehensive analytical framework rooted in proven methods of dynamical systems and advanced computational techniques. By integrating wavelet transforms with autoencoders, followed by predictive modeling using Long Short-Term Memory (LSTM) networks and dimensionality reduction via t-distributed stochastic neighbor embedding (t-SNE), we offer novel insights into the temporal and spectral characteristics of behavior. The use of LSTM networks to model the temporal sequences of behavioral states aims to predict future states and identify stable points within the system's dynamics. These fixed points are then mapped into a two-dimensional space using t-SNE, creating a visual landscape of behavioral basins of attraction. This visualization not only simplifies the interpretation of behavioral dynamics but also reveals the underlying structure and transitions between states, highlighting areas of stability and potential pathways for state changes. Our findings highlight the stability and fluidity of behavioral states, providing insights into the mechanisms governing behavioral transitions. The identification of basins of attraction and the hierarchical organization of behaviors suggest that complex behaviors may be constructed from simpler, foundational actions.

The thesis successfully demonstrates how the TACI methodology and the behavioral states pipeline provide an extensive strategy to understand dynamical systems. Together, they offer novel insights into the behavior and causality within these systems, highlighting the fluidity and stability of behavioral states, and providing a

deeper understanding of the mechanisms driving transitions. This unified approach not only advances our understanding of individual systems but also offers a broader perspective on the temporal interactions that shape the complexity of the natural world, as exemplified by its application to diverse datasets including climate patterns, neural activities in monkeys, and the behaviors of rats.

Dynamical Inference and Representations in Complex Biological Systems

By

Josuan Calderon
B.S. in Physics, Florida International University, FL, 2016
B.S. in Mathematics, Florida International University, FL, 2016

Advisor: Gordon J. Berman, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in Physics
2023

Acknowledgments

Thank you, Gordon, for everything. Being a part of this lab and witnessing its growth and evolution has been an incredibly enriching and inspiring journey. Your guidance and support have been instrumental in not only my academic and professional growth but also in fostering a lab culture that values collaboration and innovation. A special shoutout to Kanishk and those who joined the lab journey alongside me; your enthusiasm and dedication have significantly contributed to creating a supportive and collaborative environment. Our late-night discussions and weekend brainstorming sessions have not only propelled our projects but also deepened our camaraderie, making even the most challenging tasks enjoyable. Your dedication and enthusiasm towards our collective goal of making a meaningful impact through our research have truly made this experience memorable.

To my incredible family—my parents, my beloved wife, and my brother—your unwavering belief in me, especially during times when I doubted myself, has been my greatest source of strength. You pushed me beyond my perceived limits, encouraging me to pursue my dreams with relentless determination. Your support has been the cornerstone of my journey and a constant reminder of the strength I possess within. Your faith, love, and encouragement have illuminated my path, guiding me towards achieving this monumental milestone. This achievement is not just a reflection of my efforts but a testament to the enduring support and belief you have instilled in me. To my friends, who have been there for me through every high and low, your encouragement and love have been invaluable. Thank you for being my rock and my inspiration. This accomplishment is as much yours as it is mine

Contents

List of Figures

List of Tables

Chapter 1

Quantifying temporal dynamics in biological time series

1.1 Introduction

Temporal relationships are essential components of the dynamical systems that we find everywhere in nature. They are present in a wide variety of systems, ranging from the smallest biological processes to the complex mechanisms driving climate change. These interactions regulate how elements within a system impact each other across different points in time. Thus, these relationships are not static but rather are continuously flowing and evolving. For instance, these interactions may manifest as feedback loops in ecological systems [1], complex patterns of gene expression in cellular processes [2], or intricate synchronizations during epilepsy episodes seen in the human brain [3].

One classic example of temporal relationships can be found in the predator-prey system involving *Didinium* and *Paramecium*. Gause first studied this system in the 1920s, which was later further improved by Veilleux [1]. Through this convoluted connection between survival and population control, temporal interactions between

the predator and its prey lead to constant fluctuations in the population sizes over time. Another interesting case is observed in the relationship among Pacific sardine (*Sardinops sagax*) landings, northern anchovy (*Engraulis mordax*) landings, and sea surface temperature (SST) measured at Scripps Pier and Newport Pier, California [4]. Several competing hypotheses have been generated to answer for the alternating patterns of dominance of sardine and anchovy observed across global fisheries on multi-decadal time scales. Some researchers argue that the species act in direct competition with each other; however, other researchers claim that this is merely a response to common large-scale environmental forces. This example showcases the complexity of temporal interactions in marine ecosystems by demonstrating how temporal dynamics can influence species populations.

As a result, we can see that quantifying and understanding temporal dynamics is a very important topic of scientific research. Only through understanding the dynamic structure of these systems can we reach an understanding of the complex interactions present in the natural world. This thesis explores the temporal interactions influencing behavioral states and causal relationships in biological organisms. Through novel computational methods and models, this research tries to map out the temporal patterns observed in these systems and discover the fundamental mechanisms that govern these patterns.

My goal in this thesis is to build towards an integrated approach that is able to combine the concepts of temporal interactions and dynamics in a way that allows us to understand the complex flow of events and behaviors in natural and artificial systems. At the heart of this investigation, I will focus on two important areas: the causal inference of temporal interactions and the characterization of behavioral states as they unfold over time. The first area looks at identifying cause-and-effect relationships within temporal data, pushing the limits of traditional causal analysis by introducing innovative methods to capture the dynamic nature of these interactions.

The second area involves exploring the diverse range of behavioral states and mapping out the transitions and stability of these states to reveal the deeper structure of biological behavior. Together, these approaches shed light on the complex interplay of temporal dynamics, providing new insights into the patterns that govern the behavior of complex systems.

Causal Inference as a Key to Decode Temporal Interactions:

Real-world signals are usually not stationary and well-behaved, with their causal linkages and interactions often exhibiting unpredictable patterns: frequently appearing, changing, disappearing, and reappearing. Despite these challenges, many important potential applications exist for casual discovery across many fields. For instance, causality is used in climate science to comprehend the complex interactions between the many factors that affect climate change, such as how human behavior impacts global warming [4]. In healthcare, accurate detection of causal relationships is essential for developing treatment plans that can monitor the influence of different medications or lifestyle decisions on the health of the general population [5]. Moreover, economists apply causal models as a prediction tool to anticipate how various economic indicators are going to be affected by changes in policy or recent news developments [6, 7, 8]. In neuroscience, causal models have been used extensively to model the complex web of interactions between different brain regions, helping to comprehend and, in some cases, prevent neurological disorders, including Alzheimer's and Parkinson's [9, 10].

The study of temporal interactions focuses on reaching a deep understanding of a system's causal connections and relationships as they unfold over time. However, in many circumstances, identifying causality can become a difficult and arduous process due to the inherent characteristics that define these systems of interest. One of the main challenges many causal inference methods face is the non-linear relationships

observed in these systems. Small perturbations to one time series can trigger disproportionally large effects on another time series. These dynamics, unfortunately, can mask the underlying cause-and-effect mechanisms present due to the unpredictability and chaos that come with the changes from perturbing some of the variables as they ripple through the other parts of the system. Another challenge is that, in general, these systems typically contain many different interacting components that can introduce multiple layers of complexity to their analysis. As the dimensions of the data increase, the interactions become more intricate and harder to predict. Additionally, when statistical properties such as the mean and variance are non-stationary, it can further complicate matters. The presence of hidden factors, or latent variables, can also be detrimental to causal discovery since they are not directly observable through conventional methods. These hidden influences create misleading connections between variables that may appear related, leading to inaccurate detection of bidirectional causal links. Moreover, the behavior of these systems can also be affected by both internal and external environmental factors that turn the task of isolating causal relationships into an even more difficult process.

Two main approaches have been used to identify causal relationships [2]. The first involves directly tracking how perturbations to one element of a system can affect other aspects of it. However, implementing this approach can be difficult since it is often limited by ethical, logistical, and technical constraints. Examples of the approach are prominent in disciplines such as human physiology and neuroscience [11, 12, 13]. There, we see that direct manipulation of variables related to physiological processes or to those that control the flow of information between different brain regions often creates ethical concerns or may simply be unattainable by current methods of study. Therefore, on top of the intricate nature of these systems, the potential for negative consequences makes interventions in real-world systems often impractical [14]. This problem, however, is not exclusive to the life sciences. For

example, in climate research, variables cannot be treated in the same way we are used to in controlled experiments due to the magnitude of the system. Therefore, manipulating weather systems to observe causal relationships is not a viable process. Instead, researchers often develop complex models and run complex computer simulations to investigate how changes in one meteorological feature can impact other parts of the climate system [15, 16, 14].

On the other hand, if we are unable to directly perturb the system of interest, we must then rely on inferring causal interactions using a different alternative. By passively observing the features of the system as they naturally unfold over time, we can attempt to estimate the causal relationships that govern it. Inferring causality from passive observations, however, also comes with its own set of challenges. For instance, without the ability to control the system and randomly assign conditions, it is difficult to rule out alternative explanations for observed relationships. Moreover, the presence of co-founding and latent variables can obscure true causal links [17]. Thus, observed changes may be influenced by these unmeasured or unknown factors, complicating the causal interpretation. Still, this method is extremely attractive in various fields of research like genomics, ecology, epidemiology, space physics, clinical medicine, and neuroscience. The use of interventions or randomized experiments in these disciplines is often out of reach due to prohibitive costs and impractical time requirements [2].

What have researchers done given these constraints? Many methods, such as directed coherence and partial directed coherence, have been developed to tackle some of these problems [18]. Even in the most straightforward nonlinear systems, however, variables interacting for long periods can change their behavior and become anti-correlated [19]. Therefore, it makes sense to initially approach these issues using cross-correlation-based techniques. However, this method has two significant drawbacks: first, the requirement for suitably large data sets, and second, the inability to detect

nonlinear interactions [20]. Data scarcity has been less of an issue over the past years due to advances in data collection techniques. However, inaccuracy in the detection of non-linear relationships has made researchers look for better alternatives.

A more effective method of determining causality involves the mutual prediction of a few chosen observable metrics through multivariate models. In the context of stochastic linear regression models, the idea of causality, first put forth by Wiener and later more precisely defined by Granger [21], has come to be accepted as a tool for identifying directed interactions between interconnected systems. The core principle of this technique is that if signal X affects signal Y, then utilizing both Y's historical data and X's historical data should improve the prediction of Y's current state compared to simply using Y's historical data.

Many methods built on this concept, such as Granger Causality(GC), have been created to address multivariate linear models and nonlinear systems successfully. In contrast to linear or nonlinear parametric models, alternative approaches, such as Conditional Mutual Information (CMI), provide a model-independent mechanism to assess nonlinear causality, applicable to deterministic and stochastic systems [22]. Using a somewhat different strategy, Convergent Cross Mapping (CCM) operates in the state spaces of dynamic systems. The procedure continues to consider the observable variables X and Y. However, according to CCM, a deterministic dynamic system's states eventually converge on an attractor, which could be a point of equilibrium, a limit cycle, or a higher-dimensional chaotic attractor [23]. These techniques frequently require a careful assessment of the particular properties of the system to be able to detect causality in a consistent, understandable manner. These methods claim to offer solutions to most problems, but in the end, they only provide solutions to suitably selected examples. Finding a formula that will solve all the problems is unrealistic, but there should not be conflicting results with the same method for different systems with the same data type. This experience made us question the

reliability of some of these techniques and where their limitations were. Nonetheless, these methods helped us to understand the need for a mechanism that addressed continuous coupling and synchronization in non-separable noisy systems.

Since Granger causality was initially designed for linear systems, its success in nonlinear contexts can vary, depending on the problem. Even though a rather general class of covariance stationary multivariate processes—rather than just stochastic processes produced by a linear autoregressive scheme can be modeled as Vector Autoregressive (VAR) models, Granger causality can occasionally accurately capture interaction patterns in various nonlinear time series. However, there are instances where the method falls short [24]. Others have extended the effectiveness of this method by incorporating Fourier transform surrogates. This concept provides a more objective evaluation of the results by including an extra layer of testing. After fitting an autoregressive model to the time series, comparisons can be made about the variation in y, which can be explained by the addition of x and the one explained by the addition of a surrogate (x_s). As a result, we are able to calculate the Granger Causality for the surrogates using the same methodology as for the original data. With this addition, we are now able to identify powerful interactions like phase synchronization or generalized synchronization. However, the GC surrogate estimation is still limited in more complex systems that show strong coupled synchronization and often measure misleading patterns of causality that generate a substantial number of false positives.

In Chapter 3, we use this concept of surrogate Granger tests and develop a hybrid model that aims to capitalize on the strengths of this technique and new machine learning algorithms to provide more precise and trustworthy causal inferences to nonlinear problems. The framework we created does not aim to disprove many effective systems where these methods excel but rather to study a class of systems that are not covered to our knowledge by any of these more conventional methods. Our method,

called Temporal Autoencoders for Causal Inference (TACI), consists of a two-headed Temporal Causal Network (TCN) autoencoder that uses two symmetrical encoder-decoder networks, where the input sequence is encoded into a latent space and is then used to reconstruct a future representation of the original input learning more complex features in the process. TACI is deeply rooted in the Granger causality concept but addresses some of the original method's limitations to draw reliable conclusions regarding causal links.

We first illustrate the fundamentals of our methodology using simple models of stationary synthetic data, where the actual underlying dynamic system is known to us. Despite variations in signal timescale and noise, our method consistently and reliably measures the interactions between observed nodes as well as the dynamic complexity that results from these interactions.

Interpreting Temporal Dynamics through Behavioral States:

While temporal interactions are concerned with how different factors might affect one another over time, the study of temporal dynamics is centered on detecting and comprehending change patterns within a system as time advances. These efforts are less about the direct influence of variables and more about observing the evolving patterns, such as fluctuations, cycles, trends, and rhythms that characterize the evolution of variables or states. Temporal dynamics are seen in complex and non-linear systems, often resulting in unexpected behaviors and the emergence of new properties. Therefore, this concept helps us capture and decode a system's sequential behavior over time. It reveals the system's response and adaptation to both internal factors and external stimuli while simultaneously adapting and maintaining an equilibrium.

This concept plays a fundamental role in biological systems. For instance, a powerful demonstration of how temporal dynamics manifest within natural systems is the seasonal migration of the monarch butterfly (*Danaus plexippus*) [25]. These

butterflies embark on an extensive journey spanning thousands of miles from North America to Mexico each year. This system is driven by several internal and external factors that are continuously changing its dynamics. Some of these factors include environmental signals like temperature and daylight duration, alongside the internal biological instincts responsible for navigation. Another compelling example of temporal dynamics is the mating dance of the fruit fly, *Drosophila melanogaster* [26]. This courtship ritual is a complex sequence of behaviors exhibited by the male. This phenomenon includes unique alternating patterns of wing vibrations to produce courtship songs. This initiative mating behavior is followed by licking and constant attempts of copulation. Although these actions might appear random, if analyzed in a deeper context, we can see that they are part of a highly structured and timed strategy influenced by both internal genetic programming and external sensory cues from the female and the environment.

Consequently, it makes sense to approach animal behavior from the point of view of multiscale temporal dynamics. Understanding the dynamic structure of behavior is crucial for illustrating how behaviors evolve, adapt, and manifest in a hierarchical and structured manner. Numerous research studies have been published with the goal of revealing stereotypical behaviors in various organisms [27, 28, 29, 30]. The idea of stereotypy specifies that an organism's actions can be broken down into discrete, reproducible elements. For instance, Berman *et al.* [27, 28] found that most of these behavioral states correspond to familiar movements such as walking, running, front leg grooming, and proboscis extension. These states emerged from the data themselves and are not a direct consequence of *a priori* definitions. Moreover, Stephens *et al.* [30] also highlights this idea of behavior states. In this study, by analyzing the equation of motion of the nematode *C. elegans*, they discover that the space of natural worm postures can be fully described with a set of multiple attractors.

These methods rely on state-of-the-art machine learning and computer vision al-

gorithms to gain insights into the complexities of animal behavior. Even though there are subtle differences in these approaches, the primary goals remain the same: recognizing, classifying, and measuring the stereotypical actions displayed by animals in their everyday lives. The general pipeline of these processes usually starts with extracting simple postural time sequences from a set of data and then converting these still postures into dynamic representations that more accurately depict stereotypical behavior as actions rather than just poses.

A commonality between these approaches, however, is that they rely on understanding behavior through the lens of a single or a relatively small number of time scales that are semi-arbitrarily chosen [31]. Consequently, we still need a robust analytical framework that is able to isolate important behavioral dynamics across multiple time scales.

Fortunately, innovative methodologies have been created in recent years to decode the complexity of temporal dynamics, particularly using artificial neural networks [32, 33]. I will describe our attempts at adapting them to the discovery of multi-time-scale stereotyped behavioral states and temporal dynamics. In Chapter 4, we used the detailed observation of 3D kinematics in rats over extended periods for generating 2D maps that identify behavior primitive states. We found that these states act as the building blocks from which complex behaviors are constructed. The training of sequence-to-sequence Recurrent Neural Networks (RNNs) on these data enriches the analysis of behavioral states and their transitions. These types of networks allow the identification of fixed and slow points within the RNN's phase space. Once we have found these states, we can also examine transitions between them and how these transitions are influenced by temporal dynamics. The basin-like structure that emerges from this analysis is represented in a complex landscape of behavioral possibilities. Each basin denotes a distinct behavioral state. Therefore, we can think of transitions between these basins as the temporal evolution of behavioral states

over time. These transitions are not abrupt but flow in a continuum, influenced by the organism's previous experiences, its current state, and its anticipation of future needs.

1.2 Causality

As stated in the introduction to this chapter, there have traditionally been two main approaches used to identify causal relationships from dynamically changing systems. The first is concerned with the effects that actively changing one aspect of a system has on the others, while the second relies on passively observing the system's characteristics as they naturally evolve by themselves, without being affected by any external factor [2]. Causality is a keystone in our endeavor to comprehend the world around us. It is not just about observing that two events occur simultaneously; it is also important to comprehend how one affects the other and vice versa.

The idea behind causality is that one event (cause) considerably shapes another (effect). For example, if we take two distinct occurrences, A and B, in which B is the result of A, then A must exist in order for B to manifest. However, B's existence does not always mean that A must occur. In other words, the cause and effect have a domino effect relationship in which the cause influences the effect to some extent. Interestingly, this dynamic can be a two-way street, in which the outcome can sometimes loop back and influence the cause as well. Furthermore, the causal relationships are complex, where influences can follow a "many to one" or "one to many" direction. For instance, an event can be affected by several causal factors that have occurred in the past and have some influence on its present. On the other hand, one event can serve as a causal factor for several other events. As a result, studying causality requires looking at the relationships and influences between many variables and events.

It is important to note that causality is a rather wide term that applies to many different domains and applications, offering guidance and insights into decision-making. It includes science, philosophy, healthcare, economics, social sciences, and several other quantitative disciplines that aim to clarify how and why specific results are the direct product of possible cause-and-effect relationships in observational data.

Environmental scientists use causality to comprehend the complex interactions between the many factors impacting climate change, such as how human behavior affects global warming [4]. In healthcare, understanding causal linkages can be essential for developing treatment and preventative plans that can monitor the influence of medication or lifestyle decisions on health outcomes [5]. Economists apply causal models to understand how various economic indicators are affected by changes in policy or recent news [6, 7, 8]. In epidemiology, causality is crucial for monitoring the transmission of illness and developing public health initiatives [4]. In neuroscience, causal models have been used to model the complex web of interactions between different brain regions, shedding light on the etiology of neurological disorders such as Alzheimer's and Parkinson's [9, 10]. In engineering and technology, it serves as the basis for system design, AI growth, and the development of machine learning algorithms. Across all of these fields, the use of causality is integral to innovative thinking and problem-solving, offering a foundation for comprehending and navigating the intricate web of cause and effect.

1.2.1 Challenges in Causal Inference

Detecting causality in complex systems is far from a closed topic [17]. Competing hypotheses emerge in many research papers as scientists try to explain systems in which correlation, coupling, and synchronization continuously change the causal relationships of their variables.

Mirage correlations can appear in even the most straightforward nonlinear sys-

tems [34]. Variables that may be positively correlated at some point can become anticorrelated some moments after or even lose all coherence, even when the underlying dynamics driving them do not change. For instance, in fisheries, due to the changing environment that surrounds them, they can display changes in correlation [4]. Another, more concrete, example is the case of a bidirectional two-species model where two coupled logistic differential equations exhibit chaotic behavior:

$$\begin{aligned} X(t=1) &= X(t)\left[3.8 - 3.8X(t) - 0.02Y(t)\right] \\ Y(t=1) &= X(t)\left[3.5 - 3.5Y(t) - 0.1X(t)\right] \end{aligned} \tag{1.1}$$

Although correlation – even in its nonlinear form, like in information theory – is often an interesting and useful quantity to measure, the presence of correlation is neither necessary nor sufficient to demonstrate causation (and the same is true for the lack of correlation) [35, 36]. But just as correlation changes with time, variables can show alternating patterns of dominance. Time-varying coupling linkages have been widely investigated in finance. The coupling fluctuations tend to behave strongly during periods of financial turmoil and debt crisis [8, 37, 6]. Understanding the essence of these linkages is vital to making appropriate risk management decisions and increasing returns.

In biology, even though observational error, noise, and limited data affect the accuracy of any test, coupling changes in the time domain are still significant. For example, time-varying coupling was found between two EEG signals during a typical absence seizure from patients with childhood absence epilepsy [3]. In ecology, some hypothesized that in the case of landings of Pacific sardine and northern anchovy, the variables of the system exhibit a varying coupling due to the nature of the direct competition between the species [38]. As one population starts to peak, the other declines.

At certain coupling values, the possibility of correctly predicting the driver-response

relationships between systems remains deeply ingrained in the concept of generalized synchronization. We interpret generalized synchronization between variables X and Y as the following:

$$y(t) = F\left(x(t)\right) \tag{1.2}$$

where F is a transformation that describes how these variables adjust their dynamics to a common behavior due to the coupling of the system [39].

We differentiate between two primary processes that might result in synchronized states. First, consider the case of unidirectional coupling, while X evolves independently and freely but simultaneously influences and controls the state of Y. Therefore, the response subsystem becomes enslaved, closely mirroring the behavior of the master subsystem. Second, there is the case of bidirectional coupling, where the relationship is reciprocal with no slaves or masters. Here, both X and Y are coupled with each other, thus creating a rhythm adjustment that results in mutually synchronized behavior.

These situations have become an area of active study in a range of scientific fields. For instance, investigations about the extreme events of the coupled ocean-atmosphere phenomenon *"El Niño/Southern Oscillation"* have shown that internal synchronizations play a crucial role in phase-to-phase causal linkage [40]. More examples can be found in physiology, e.g., the human cardio-respiratory system, where a long period of synchronization was discovered during breathing, even for weak coupling between respiration and cardiac rhythm [41].

My goal in Chapter 3 of this thesis is to quantify the extent of time-varying coupling on the causal linkage between variables of the interacting systems. I accomplish this goal using a novel approach that is based on temporal convolutional network autoencoders and a new metric for assessing causality using a surrogate data approach.

1.3 The Dynamics of Behavior

The concept of behavioral dynamics in biological studies is complex and has been described in several ways. These concepts often reflect the complexity and variety of actions and responses exhibited by organisms [42]. These ideas are broad and include things like "all actions performed by an organism," while others are more focused and include things like "a total of movements made by the intact animal" [43].

Some definitions of behavior focus on the relationship that exists between an organism and its environment. This definition allows one to see the behavior as a dynamic response to environmental stimuli from external factors. In contrast, alternative interpretations place emphasis on the internal processes of the organism, thus suggesting that behavior is primarily dictated by internal states and physiological processes. A more integrative approach takes into account both external and interior influences. It views the behavior as a result of the interaction between an organism's internal state and its external environment.

1.3.1 Ethology and Tinbergen's Four Questions

The biological study of animal behavior is known as ethology, and it includes a wide variety of topics such as functional, phenomenological, causal, and ontogenetic. These areas of research, also known as "Tinbergen's Four Questions," offer a comprehensive approach to examining animal behavior [43, 44].

- Function (Adaptation): Why does the animal engage in this behavior, and how does it increase its overall fitness? This question asks the adaptive value of a behavior. It aims to understand the role that behavior plays in enhancing the animal's survival and reproductive abilities.

- Evolution (or phylogeny): How has the behavior evolved over time due to the influence of natural selection? This question investigates the evolutionary history

of behavior. It is a comparative approach where the behavior among closely related species is analyzed to determine how the behavior may have evolved from the animal ancestors.

- Causation (or mechanism): What physiological mechanisms trigger the execution of behavior? This question addresses the immediate reasons for the behavior. It involves understanding the stimuli and environmental factors that cause a particular behavior.

- Development (Ontogeny): How have experience and learning shaped the development of an individual's behavior over the course of their lifetime? This question focuses on how a person's behavior changes from birth until adulthood. It pays special attention to the influence of genes, learning, and environmental triggers on the development of behavior.

The field of animal behavior, as outlined in the influential work of Niko Tinbergen in 1951 and 1963 [43, 44], originates from the groundbreaking studies of Konrad Lorenz, Karl von Frisch, and Niko Tinbergen themselves, who were collectively awarded the 1973 Nobel Prize for their notable contributions to our understanding of animal behavior.

The primary focus of ethology is to provide an in-depth description and characterization of behavior. This approach is crucial for understanding behavior within its ecological and evolutionary framework. Initially, ethological studies were predominantly qualitative. They relied on detailed observational descriptions of animal behavior. Ethologists would primarily observe animals in their natural habitat and record their findings about animal life and behavior.

However, over the past decades, as new techniques for recording and describing behavior have developed, there has been a significant shift towards more quantitative methods in ethology. This transition involves the use of systematic methods to observe

and document specific behaviors based on well-defined criteria. This approach allows the use of statistical tools to interpret the numerical data extracted from animals' behavior. For example, an observer can document the frequency of a particular behavior, the time it takes for an animal to initiate that behavior (known as latency), and the duration of each instance of the behavior and compare the frequency or duration of various behaviors. These quantitative evaluations have introduced fresh opportunities in ethological investigations, enabling more rigorous assessments of theories regarding animal behavior, its origins, progression, and evolutionary importance.

1.3.2 Measuring Behavior

Measuring behavior is a critical aspect of psychological, biological, and ethological research. Even with modern techniques, it has proven to be a challenging task due to the complexity and variability of behaviors. Behaviors must be defined clearly and objectively to ensure that what is being measured is consistent and reliable [45].

Historically, researchers have analyzed behavior using different approaches. A widely used technique was the paradigmatic approach, where behavior quantification is integrated into the experimental setup. While this method may not fully encompass the complexities of behaviors, it does provide a simplified, low-dimensional perspective, although in some cases, potentially unnatural measurement [46]. Another method involves measuring more coarse variables. However, this tends to focus on behavior at a singular scale and may overlook intricate behaviors like specific grooming patterns or subtle social interactions. More recently, researchers have developed human-defined classification systems and scored by trained observers. This approach allows for a deeper, more detailed understanding of behavior but is extremely labor-intensive. Still, this process is time-consuming and requires considerable human resources. Moreover, the subjective nature of these systems makes it difficult to argue quantitatively that one representation of behavior is more accurate or appropriate

than another, limiting reproducibility across studies [31]. To overcome these obstacles, new technological advancements have been created. The use of automated tracking systems, machine learning algorithms, and advanced statistical methods have allowed scientists to quantify and analyze behavior more objectively and efficiently.

1.3.3 Stereotyped Behavior

A great deal of modern biological research is focused on studying stereotyped behaviors in animals (defined in the previous section as movements an animal makes that are performed often and repeatably). As a result, powerful analytical methods have been developed to investigate these patterns without the requirement for manually labeled examples. This transition towards data-informed, unsupervised analysis is based on the idea that a considerable portion of their behavior is low-dimensional and exhibits repetitive patterns. Therefore, this implies that animals tend to rely on a limited repertoire of movements that are repeated on a consistent basis under similar contexts or in response to certain stimuli.

Several studies have developed numerous tools to uncover and understand these stereotyped behaviors across a broad spectrum of organisms [27, 29]. The methodologies employed in these studies are diverse, encompassing various techniques from machine learning and computer vision to neuroscience and ethology. However, despite all the differences in these approaches, these studies share the same objectives: to identify, categorize, and quantify the stereotyped behaviors exhibited by animals during their daily activities.

In general, these approaches to studying animal behavior typically begin by extracting low-dimensional postural time series from a dataset. Traditionally, this reduction to low-dimensional data has been accomplished by tracking specific parts of the animal's body, such as joints, leg tips, tails, or heads.

The next stage, once these postural data have been obtained, is to convert these

static postures into a dynamic representation. This transformation is crucial because when we define stereotyped behavior, we associate movements and not postures. For example, walking is not simply described by the specific angles of knee or ankle joints at a given moment but by the trajectory of these angles over time [31]. Therefore, creating a dynamic model that accounts for the way these postures alter over time is essential. One way of achieving this is by fitting a differential equation directly to the postural data or by identifying dynamic features within segments of the data. Another method for representing behavioral dynamics across multiple time scales is time-frequency analysis [27, 28]. This allows us to measure the importance of the frequencies present in each series over time. Time-frequency analysis offers a multifaceted view of the animal's behavior by showing how certain movements or behaviors manifest across different temporal scales.

The primary objective of these procedures is to create a behavioral representation that can identify the longer-term changes in the underlying postural movements that give rise to the observed motions. An ideal dynamical representation would naturally emerge from the analysis of postural dynamics, capturing the essence of the animal's behavior in a way that is both comprehensive and understandable.

1.4 Thesis Outline

This thesis is divided into five chapters. While this chapter serves as a broad introduction, **Chapter 2** serves to provide a background of the analytical tools and their underlying theories that are discussed throughout the rest of this thesis. Specifically, I have provided a comprehensive overview of foundational concepts and methodologies in the fields of temporal dynamics, causality, and behavioral state analysis. Beginning with an exploration of both linear and nonlinear correlations, this section delineates the difference between correlation and causality, setting the stage for a

deeper investigation into causal interactions. The chapter then moves on to outline established methods for deducing temporal causal connections, including Granger causality, Transfer Entropy, and Convergent Cross Mapping. Additionally, it introduces key concepts of dynamical systems, such as basins of attraction and fixed points, along with the application of recurrent neural networks (RNNs), particularly LSTM models, for the analysis of intricate temporal sequences. The chapter concludes with a review of methods for behavioral analysis, emphasizing wavelet transform, autoencoders for dimensionality reduction, and spatial embedding techniques as pivotal tools in capturing the essence of behavioral states.

Chapter 3 introduces Temporal Autoencoders for Causal Inference (TACI), a novel method created to tackle the complexities of causal relationships within dynamic systems. Traditional approaches to causal inference often fall short in handling the nonlinear, non-stationary behaviors and varying strengths of causal links in real-world variables. TACI addresses these challenges using a dual-headed Temporal Convolutional Network (TCN) autoencoder structure, which leverages the strengths of TCNs—simplicity, long-term memory retention, and auto-regressive prediction capabilities—to analyze time-series data effectively. Its efficacy was demonstrated across deterministic and stochastic models, as well as in varied practical scenarios, including artificial models like Autoregressive Models and Henon Maps, to investigate coupling strength effects and dynamic causal relationships. Moreover, we explored its practical applicability using real-world datasets, including the Jena Climate Dataset and electrophysiological data from a monkey during different states of consciousness.

In **Chapter 4**, the focus shifts to the representation of behavioral states through dynamical models. The introduction sets the stage for a detailed exploration of behavioral states and the significance of RNN fixed points in modeling these states. The chapter demonstrates how dynamical models may capture the emergent features of behavior through the examination of postural decomposition, spectrogram gen-

eration, and the discovery of latent dynamics and fixed, and slow transition states. The discussion on embedded space dynamics and transition matrices highlights the model's ability to clarify the hierarchical organization of behavioral states, emphasizing the predictability and hierarchy inherent in animal behavior.

Chapter 5 reflects on the thesis's contributions to the fields of temporal dynamics and behavioral analysis, summarizing the key findings of each chapter and discussing the implications of this work. It acknowledges the limitations encountered, particularly in the application of TACI for causal inference and the fixed points pipeline in modeling dynamical systems. Looking forward, the chapter outlines promising avenues for future research, such as investigating brain connectivity in prairie voles during social bonding and exploring the concept of brain states.

Chapter 2

Background Information

2.1 Introduction

In this thesis, I introduce novel approaches to quantifying temporal dynamics in biological time series. To accomplish this aim, I developed two methods: the first analyzes the temporal interactions within these complex systems, and the second aims to decode a representation of the temporal dynamics of behavioral states. Accordingly, I have divided this chapter into two parts, each corresponding to the technical background information that underlies these two methods, respectively.

The chapter begins by exploring the different methods used for discovering temporal causal links, such as Granger tests, Transfer Entropy, and Convergent Cross Mapping, as well as those based on Artificial Neural Networks. The second part of the chapter then ventures into the realm of dynamical systems, highlighting key concepts, such as Recurrent Neural Networks and fixed points, that are used for analyzing the temporal dynamics of the intricate patterns of behavioral states.

2.2 Methods for Causal Inference Analysis

2.2.1 Correlation

Correlation is a statistical tool that characterizes the relationship between two or more variables. In statistics, "correlation" often refers to the degree to which two variables vary together; however, in the broadest meaning, it can represent any kind of link. In essence, it measures the relationship between two or more variables. It is an essential concept in the domains of statistics and data analysis that sheds light on the connections between various data sets.

Linear Correlation

The most often used indicator to measure the degree of dependency between two quantities is "Pearson's correlation coefficient." It is a methodical process that quantifies the linear relationship between two variables by calculating the ratio of the covariance of the two variables in issue and dividing by the product of their standard deviations. The objective of the correlation coefficient is to determine an optimal line that best fits the dataset comprising the two variables. It does this by comparing the expected values to the actual data points. The Pearson's correlation coefficient results show how much the actual dataset deviates from the predicted values.

The Pearson's correlation coefficient $\rho_{X,Y}$ between two random variables X and Y with expected values μ_X and μ_Y and standard deviations σ_X and σ_X is defined as:

$$\rho_{X,Y} = \operatorname{corr}(X,Y) = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{\mathrm{E}\left[(X-\mu_X)\,(Y-\mu_Y)\right]}{\sigma_X \sigma_Y}, \quad \text{if } \sigma_X \sigma_Y > 0, \quad (2.1)$$

where μ_X and μ_Y are the expected values and σ_X and σ_X are standard deviations. The E is the expected value operator, cov stands for covariance, and corr is a commonly used abbreviation for the correlation coefficient. Only finite and greater-than-zero

standard deviations for both can be used with the Pearson correlation. This calculation also assumes that both variables come from a normal distribution.

The Pearson's coefficient, which ranges from -1 to +1, indicates the direction and intensity of the linear relationship between the variables under investigation. Strong positive correlations are indicated by coefficient values that are close to +1, whereas strong negative correlations are indicated by values that are close to -1. A number close to 0 denotes a very weak linear connection.

Nonlinear Correlation

In situations where the statistical relationship between two variables does not follow a straight line, nonlinear correlation has been established as a more reliable statistical method than Pearson. Put otherwise, the rate of change in one variable does not remain constant with respect to the other. The evaluation of Pearson's Correlation Coefficient would only provide the direction and strength of the linear association between the variables of interest when there is no discernible linear relationship between two random variables but rather a monotonic relation (if one increases, the other increases or decreases). On the other hand, nonlinear approaches such as Spearmans's and Kendall's rank correlation coefficient method would provide us with the strength and direction of the monotonic relation between the connected variables. This concept is essential in various fields since nonlinear interactions are frequently observed in real-world data.

Spearman's rank correlation coefficient: Spearman's rank correlation coefficient, also known as Spearman's ρ, is a non-parametric indicator of statistical dependence between two variables. It evaluates the degree to which a monotonic function can adequately characterize the relationship between two variables, and it is based on the rankings of the data rather than their absolute values. This method makes

it perfect for continuous data that don't meet the assumptions of normality needed for Pearson's correlation or for ordinal data where the exact changes between levels aren't always significant. The robustness of Spearman's correlation to outliers and non-normal distributions is one of its main benefits. The correlation coefficient is less affected by extreme values because it is based on rankings rather than absolute values.

To calculate Spearman's ρ, each set of data is ranked independently. On these ranks, the Pearson correlation coefficient is then calculated as:

$$\rho_{R(X),R(Y)} = \frac{\text{cov}(R(X), R(Y))}{\sigma_{R(X)}\sigma_{R(Y)}}, \tag{2.2}$$

where $R(X)$ and $R(Y)$ are the rank of X and Y respectively. The value ranges between -1 and +1. A perfect positive monotonic connection is denoted by a coefficient of +1, a perfect negative monotonic relationship by a value of -1, and no monotonic relationship is implied by a coefficient of 0. Essentially, it measures the degree to which a relationship between two variables is monotonic, that is, such that as one variable grows, the other continuously increases or decreases. Spearman's correlation has drawbacks despite its versatility. It may not adequately represent the strength of non-monotonic relationships as it just measures monotonic relationships. Additionally, its meaning may become less clear in circumstances when there are several tied rankings.

Kendall's rank correlation coefficient: Kendall's rank correlation coefficient, also known as Kendall's τ, is another non-parametric statistic used to measure the association between two measured quantities. The foundation of Kendall's τ lies in the idea of concordance and discordance between pairs of observations. To put it simply, any two observations are concordant if and only if the rank order is the same in both pairs and discordant if the rank order differs. Being robust is one of

Kendall's tau's main benefits, particularly when working with datasets that have a large number of connections or small sample sizes. It is a more trustworthy metric in some situations than Pearson's correlation because it is less susceptible to outliers.

To calculate Kendall's τ, one must first count the number of concordant and discordant pairings. The difference between these counts must then be calculated and normalized by the total number of pairs [47]. This technique works especially well with data sets that have a large number of tied ranks or a small number of observations. Mathematically, this is represented as:

$$\tau = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{(\text{number of pairs})}. \tag{2.3}$$

The values for Kendall's τ oscillate between -1 and +1. A value of +1 represents a perfect positive relationship, whereas a value of -1 is a perfect negative association, and a 0 value corresponds to no association. Kendall's τ, unlike Spearman's ρ, is usually more conservative and accurate since it produces a lower value for a given dataset. This feature becomes very handy when overestimating the strength of the correlation leads to incorrect conclusions. However, these advantages also carry some limitations since they are more computationally complex and can become a limiting factor for particularly large datasets.

Correlation vs Causality

In common (non-technical) usage, people often use the terms correlation and causality as if they mean the same thing, but they actually relate to distinct ideas. The distinction between correlation and causality is a fundamental concept in various scientific fields, such as statistics, economics, epidemiology, computer science, and philosophy.

Correlation is a specific type of association that indicates a pattern or trend in the

data, such as an increase or decrease in one variable that corresponds with changes in another. The general idea of association, or dependence, states that one variable can provide information about another. Although it does not necessarily indicate a causal relationship, it does reflect that the variables are connected in some manner [48]. In a deck of cards, for example, drawing a red card affects the probability of getting a black card after it, demonstrating a link brought about by the change in the composition of the deck. Thus, correlation denotes the extent to which two or more variables fluctuate together. It can be calculated using Pearson's correlation coefficient for linear trends or Spearman's and Kendall's rank correlation for nonlinear trends, as we saw in Section 2.2.1.

Causal inference, on the other hand, goes beyond simple association or correlation. Its focus is on establishing whether the relationship between variables is causal – if changes in one variable cause changes in another. This inference involves understanding the circumstances in which particular effects occur.

There are many circumstances when it seems obvious that one action causes another, but there are also many others where it is difficult to determine and validate the nature of this link. As a result, it is crucial to understand the difference between correlation and causation in order to prevent drawing incorrect conclusions. This is exemplified in the historical case of the Aristotelian theory of spontaneous generation [49]. Although it was grounded in empirical observations after noticing that flies appeared when there was decaying meat, this theory proposed that live things may emerge spontaneously from non-living matter, a causal explanation. Concluding that the presence of flies was causally related to the existence of decaying meat represented an erroneous interpretation of the relationship. Louis Pasteur's investigations did not disprove this notion until the 19th century, underscoring the need to differentiate between correlation and causation.

2.2.2 Granger Causality

The notion of causality in experimental practice, particularly in the analysis of time series data, was significantly advanced by Clive W. J. Granger, recipient of the 2003 Nobel Prize in Economics. Granger, who first proposed this theory in 1969 [21], made significant contributions to the study of economics by putting out the argument that causality could be tested by measuring the forecast ability of a time series by using the past values of another time series. The statistical definition of this theory is inspired by Norbert Wiener's work and identifies two primary premises about causation [50]:

1. The cause occurs before the effect; and

2. The cause contains information about the effect that is unique and is in no other variable.

Granger causality, which originally developed in the field of econometric time series analysis, has been accepted as one of the most important theories since people first became interested in the field of causality [51]. The growing popularity of this method has led researchers to find applications for it in a variety of different fields. It has had a profound impact on economics and neuroscience in the last few decades. It provides a platform that uses predictability instead of correlation to identify causation between time series variables recovered from the brain signals. If X "Granger Cause" Y, then information about X will be contained in Y and cannot be removed from the universe of all possible causative variables.

A necessary condition for GC is separability. This means that information about the causative factor has to be inherent to that variable (e.g., information about the effects of the predator is not contained in the time series of the prey) and can be omitted by removing that variable from the model. Separability is a feature of purely stochastic and linear systems, which allows GC methods to be quite useful for identifying interactions between strongly coupled systems.

Granger's vector autoregressive test

In its most classical formulation, GC is generated from Vector Autoregressive (VAR) stochastic processes, which use linear regression to characterize the dynamics between the past and present states of the observable processes. This method assumes that the selected model can accurately capture the whole interaction between the systems. If this assumption does not hold true, the regression coefficients may not support the causal inferences made, which might lead to incorrect conclusions regarding causality [52]. For instance, to illustrate the mathematical formulation of this theory, let us consider two variables x and y that belong to a linear autoregressive model:

$$\begin{aligned} x(t) &= \sum_{j=1}^{p} A_j^{xx} x(t-j) + \sum_{j=1}^{p} A_j^{xy} y(t-j) + E^{x/y}(t) \\ y(t) &= \sum_{j=1}^{p} A_j^{yx} x(t-j) + \sum_{j=1}^{p} A_j^{yy} y(t-j) + E^{y/x}(t), \end{aligned} \tag{2.4}$$

where j is the number of lagged observations of the model, the matrix A contains the contributions of each lagged observation to the two variables x and y of the model, and $E^{x/y}$ and $E^{y/x}$ are the predictions errors for each variable. In the case of a bivariate VAR model of lag p we calculate the error ratio $1 - [\mathrm{var}(E^{y/x})/\mathrm{var}(E^{y})]$, where a negative ratio result indicates that x fails to "Granger Cause" y. We can define $E^{y}(t)$ as follows:

$$y(t) = \sum_{j=1}^{p} A_j^{y} y(t-j) + E^{y}(t). \tag{2.5}$$

Traditionally, when testing for Granger Causality, one uses an asymptotic $F-$ Distribution with p and $T - 3p$ degrees of freedom that compares the past values of x and y (full model) to only the past values of x (reduced model)

$$F = \frac{(RSS_{red}^2 - RSS_{full}^2)/p}{RSS_{full}^2/(T-3p)}, \tag{2.6}$$

where $RSS^2_{full} = \sum(E^{y/x})^2$ and $RSS^2_{red} = \sum(E^{y})^2$ are the sum of the squared residuals of the full and restricted model, respectively, and T is the number of observations. using this test, we can say that y Granger cause x if the result of the F-test (from Equation 2.6) is above the $(1-\alpha)\%$ quantile.

Since this causal VAR modeling was created for linear models, its application to nonlinear dynamics has several notable limitations. In some cases, it can overlook complex patterns of interactions between variables. Additionally, its coefficients explain the time-lag effects between the processes, but they do not take into consideration the instantaneous (i.e., not lagged) effects. This method requires the time series to be stationary, i.e., its statistical parameters, such as mean and variance, must remain constant over time in order to draw correct conclusions about causality.

Extended Granger test

The classic Granger causality test is formulated for linear regression models and assumes that the time series are stationary. Nonetheless, nonlinear behavior and non-stationary features are present in a wide range of real-world problems. Furthermore, this method lacks a way that can simultaneously address both instantaneous and lagged effects. As a result of this incomplete description, the traditional VAR models can generate incorrect interpretations of causation [52].

Extender Granger causality is an advanced modification of the classic Granger causality test, aiming to solve some of its drawbacks and increase its applicability in a larger variety of contexts [53]. Even for nonlinear and non-stationary data, the dynamics of the processes may still be locally approximated using simple linear regression. To achieve this, we must first generate a delay embedding reconstruction of the phase space attractors. Following this, traditional Granger causality is applied to each local neighborhood, and then the results are averaged over the entire state portrait of the dynamics.

Consider the time delay embedding vector $Z(t) = [X(t), Y(t)]$ according to Takens [54], formed by the two nonlinear time series $X(t)$ and $Y(t)$:

$$\begin{aligned} X(t) &= [x(t), x(t-\tau_X), \ldots, x(t-(d_X-1)\tau_X)] \\ Y(t) &= [y(t), y(t-\tau_Y), \ldots, y(t-(d_Y-1)\tau_Y)], \end{aligned} \tag{2.7}$$

where τ_X and τ_Y are the time delay and d_X and d_Y are the embedding dimensions for X and Y, respectively. In general, the time delays and embedding dimensions are different for different time series. However, in order to determine causation using Granger's VAR local neighborhood approximation, we must equal the time delays $\tau = \tau_X = \tau_Y$.

In the delay embedding space, there exists a function that can map a specific point $Z(t)$ to its subsequent observed image, $Z(t+\tau)$. Even if the analytical form of this function is not known, it is possible to do a local linear approximation as $Z(t+\tau) = A \cdot Z(t) + R(T)$ [55]. Here, the coefficient matrix A can be calculated using the least squares method, and the R terms represent the error vector between the actual observed values and those predicted by the linear approximation

$$\begin{aligned} \begin{pmatrix} x(t+\tau) \\ y(t+\tau) \end{pmatrix} =& \mathbf{A}_1 \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} + \mathbf{A}_2 \begin{pmatrix} x(t-\tau) \\ y(t-\tau) \end{pmatrix} \\ &+ \cdots + \mathbf{A}_d \begin{pmatrix} x(t-(d-1)\tau) \\ y(t-(d-1)\tau) \end{pmatrix} \\ &+ \begin{pmatrix} E^{x/y} \\ E^{y/x} \end{pmatrix}, \end{aligned} \tag{2.8}$$

where the embedding dimension $d = d_X = d_Y$ has been assumed to be equal to simplify the equation. Furthermore, each time series can be fitted in a neighborhood

of Z using linear regression approximations:

$$\begin{aligned} x(t+\tau) &= \sum_{j=1}^{d_X} A_j^x x[t+(j-1)\tau] + E^x(t) \\ y(t+\tau) &= \sum_{j=1}^{d_Y} A_j^y y[t+(j-1)\tau] + E^y(t). \end{aligned} \tag{2.9}$$

The error ratio can be calculated as $1 - [\mathrm{var}(E^{y/x})/\mathrm{var}(E^y)]$ by applying the concepts of Granger causality to these local linear systems. The process above is replicated across a set of selected neighborhoods distributed over the entire attractor. This is done to ensure we sample the full attractor adequately by averaging the error ratios obtained from each local region. Thus, the extended Granger causality index (ECGI) can be computed as:

$$\Delta_{y\to x} = \left\langle 1 - \mathrm{var}(E^{x/y})/\mathrm{var}(E^x) \right\rangle, \tag{2.10}$$

where the symbol $\langle * \rangle$ refers to the averaging of the error ratio over all the chosen neighborhoods. When the ratio of the errors, expressed as $\mathrm{var}(E^{x/y})/\mathrm{var}(E^x)$ or $\mathrm{var}(E^{y/x})/\mathrm{var}(E^y)$, falls below 1, it suggests that X or Y is causally influenced by Y or X, respectively.

A key challenge in this method lies in determining the optimal size of the neighborhood, denoted as δ. For the method to be effective, the number of data points in each chosen area must be large enough to guarantee statistically significant results [24, 56]. On the other hand, the neighborhood also needs to be sufficiently small to support the linearity assumption.

In linear systems, the value of this index remains constant, even as the neighborhood size δ becomes smaller. In contrast, for nonlinear systems, the index begins to show the true nature of the nonlinear causal relationships within the system. Consequently, the selection of δ is not merely a technical detail but an important decision

that can significantly impact the results of the analysis.

2.2.3 Information theory as a tool for causality detection

In the field of information theory, the primary metric for measuring the information of a discrete random variable is its Shannon Entropy [57]. This entropy quantifies the decrease in uncertainty or unpredictability present when the actual value of a variable is measured. Wiener's approach to causality is grounded in the concept of predictability improvement. In this framework, a signal X causes Y if we can better predict the future state of Y with the inclusion of both the past and present of X [58]. Therefore, if an increase in predictive power can be associated with a reduction in uncertainty, then it makes sense to describe causality in terms of information-theoretic principles [59].

Shannon entropy is a powerful measure for analyzing and interpreting data, with applications in numerous fields. It allows for a deeper understanding of the intrinsic uncertainty and unpredictability present in a wide range of processes, from simple communication systems to complex biological networks. This knowledge is essential for both theoretical research and real-world applications where precise prediction and effective data processing are critical.

Entropy and Mutual Information

The concept of entropy is essential to comprehend the nature of information. Assume X is a discrete random variable having possible outcomes $x_1, x_2, \ldots, x_n$ and their associated probability distribution $p_i = (x_i)$, where $i = 1, \ldots, n$. Then, the entropy can be explicitly written as:

$$H(X) = -\sum_{i=1}^{n} p_i \log p_i, \tag{2.11}$$

where the sum extends over all states i the process can assume [57]. The base of the logarithm, bits when the base is two and nats when it is e, determines the units used to measure the information. Let X, Y be two discrete random variables; we can define the combined or joint entropy analogously

$$H(X,Y) = -\sum_{i=1}^{n_X}\sum_{i=1}^{n_Y} p(x_i,y_i)\log p(x_i,y_i), \tag{2.12}$$

where $p(x_i,y_i)$ is the corresponding joint probability that the variable X is the state x_i and Y is in the state y_i. However, sometimes, we face a problem where the probability of an output depends on certain inputs. In this case, the joint probability can be written in terms of the conditional entropy as follows:

$$H(X,Y) = H(X|Y) + H(Y), \tag{2.13}$$

where

$$H(X|Y) = -\sum_{i=1}^{n_X}\sum_{i=1}^{n_Y} p(x_i,y_i)\log p(x_i|y_i), \tag{2.14}$$

and $p(x_i|y_i)$ is the conditional probability [60].

Now that we have quantified the total amount of information shared by a specific pair of random variables, we must adopt a different approach to determine how much one random variable reveals about the other. Mutual information (MI) of a pair of variables reflects the mutual reduction in uncertainty about one variable given the knowledge of another. Mathematically, for two random variables X and Y, mutual information can be defined as a combination of the marginals and joint probability distributions.

$$\begin{aligned} I(X;Y) &= H(X) + H(Y) - H(X,Y) \\ &= -\sum_{i=1}^{n_X}\sum_{i=1}^{n_Y} p(x_i,y_i)\log\frac{p(x_i,y_i)}{p(x_i)p(y_i)}. \end{aligned} \tag{2.15}$$

For jointly discrete pairs, the mutual information can be seen as the Kullback-Leibler divergence (KLD) between the joint distribution $p(x_i, y_i)$ and the product of the marginal distributions $p(x_i) \cdot p(y_i)$. In other words, it quantifies the deviation between the joint distribution of X and Y and what it would be if they were independent. The KLD, also known as relative entropy or cross-entropy, is used as an alternative method to mutual information [61]. It measures the excess number of bits that will be coded if an alternative distribution is chosen. For discrete probability distributions $p(x)$ and $q(x)$ of a variable X, the KLD is defined as:

$$D_{KL}(p, q) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}. \tag{2.16}$$

Mutual information is a specific application of KLD that quantifies the shared information between variables and provides a metric for the degree of dependency or association between them. For this reason, it serves as a bridge between entropy and KLD, contributing to a deeper knowledge and comprehension of information theory. Mutual information captures nonlinear as well as linear interdependence, in contrast to correlation, which only assesses linear connection.

Conditional Mutual Information

Rooted in the principles established by Shannon [57], conditional mutual information (CMI) builds upon the concept of mutual information by introducing a conditional aspect. This powerful statistical technique was introduced by Paluš *et al.* [62] to understand the causal relationship and directionality of coupling between the variables of interest. Thus, the CMI $I(y; x_\tau|x)$ and $I(x; y_\tau|y)$ allows us to identify the direction of the link between the processes $X(t)$ and $Y(t)$. A causal link is indicated by a significantly high value of the CMI between the variables studied.

Let $x(t)_{t=1}^{M}$ and $y(t)_{t=1}^{N}$ be two finite time series that represent the evolution of two

dynamical systems in spaces of dimensions M and N, respectively. Then, according to Takens [54], we can use the time delay coordinates instead of the original components of the vectors. As a result, the direction of coupling can be calculated as:

$$I(\vec{Y}(t); \vec{X}(t+\tau)|\vec{X}(t)) = I((y(t), y(t-\rho), \ldots, y(t-(M-1)\rho)); \\ x(t+\tau)|x(t), x(t-\eta), \ldots, x(t-(N-1)\eta)), \qquad (2.17)$$

where η and ρ are the lags used for the time delay embeddings of the processes $X(t)$ and $Y(t)$, respectively. The $I(\vec{X}(t); \vec{Y}(t+\tau)|\vec{Y}(t))$ can be defined analogously. This method can be understood as an information-theoretic formulation of Granger causality [51]. A high CMI value is an indication of an information flow between $\vec{Y}(t)$ and $\vec{X}(t+\tau)$ conditioned on the past of $\vec{X}(t)$. Since the forward time lag, τ, is typically unknown beforehand, the CMI is calculated as a function of τ. Consistent findings across various forward time lags, τ, after averaging imply the existence of a causal link from $Y(t)$ to $X(t)$.

Transfer Entropy

Transfer entropy (TE) was developed by Schreiber *et al.* [22] as a metric to quantify the transfer of information between temporally evolving systems. The concept of mutual information can be adapted to have a directional aspect by adding a time lag into one of the variables and then computing it

$$M_{IJ}(\tau) = \sum p(i_n, j_{n-\tau}) \log \frac{p(i_n, j_{n-\tau})}{p(i)p(j)}. \qquad (2.18)$$

This method allows the inclusion of a dynamic structure by shifting the focus from static probabilities to transition probabilities. For instance, in a system that can be approximated by a stationary Markov process of order k, the conditional probability of finding I in a particular state, i_{n+1}, at a given time $n+1$ is independent of the state

i_{n-k}. Therefore, the transition probabilities describing the evolution of the system are

$$p(i_{n+1}|i_n, \ldots, i_{n-1+1}) = p(i_{n+1}|i_n, \ldots, i_{n-1+1}, i_{n-k}). \tag{2.19}$$

Assuming all prior states are known, the entropy rate can be used to determine the average number of bits required to encode one additional state of the system. Calculating this involves comparing the Shannon entropies of the processes represented by the k and $k+1$ dimensional delay vectors constructed from I

$$h_I = -\sum p(i_{n+1}, i_n^{(k)}) \log p(i_{n+1}, i_n^{(k)}). \tag{2.20}$$

However, it is preferable to generalize the entropy rate to more than one system for the study of the dynamics of shared information. The most intuitive way we can approach this idea is by expanding h_I to a mutual information rate that measures the deviation from independence between two processes. The generalized Markov property, assuming that I and J are independent is:

$$p(i_{n+1}|i_n, \ldots, i_{n-k+1}) = p(i_{n+1}|i_n^{(k)}, j_n^{(l)}), \tag{2.21}$$

where k and l are the orders or conditioning states from processes I and J respectively, in such a way that $i_n^{(k)} = (i_n, \ldots, i_{n-k+1})$ and $j_n^{(l)} = (j_n, \ldots, j_{n-l+1})$. Thus, because of its symmetry, Schreiber proposed using the Kullback-Leibler divergence for quantifying the difference in the transition probabilities from the generalized Markov property (Eq. 2.21) [60].

A lack of information flow from J to I makes the state of J have no impact on the transition probabilities of I. The additional bits that must be employed to encode the information about the process's state can be measured by a KLD, by which we

define the transfer entropy as:

$$T_{J\to I} = \sum p(i_{n+1}|i_n^{(k)}, j_n^{(l)}) \log \frac{p(i_{n+1}|i_n^{(k)}, j_n^{(l)})}{p(i_{n+1}|i_n^{(k)})}. \tag{2.22}$$

However, using some of the properties of joint and conditional probabilities, we can manipulate Eq. 2.22 to obtain:

$$\begin{aligned} T_{J\to I} &= \sum p(i_{n+1}, i_n^{(k)}, j_n^{(l)}) \log \frac{p(i_{n+1}, i_n^{(k)}, j_n^{(l)})}{p(i_{n+1}|i_n^{(k)}) p(i_n^{(k)}, j_n^{(l)})} \\ &= \sum p(i_{n+1}, i_n^{(k)}, j_n^{(l)}) \log p(i_{n+1}, j_n^{(l)}|i_n^{(k)}) \\ &- \sum p(i_{n+1}, i_n^{(k)}) \log p(i_{n+1}, i_n^{(k)}) - \sum p(i_n^{(k)}, j_n^{(l)}) \log p(i_n^{(k)}, j_n^{(l)}). \end{aligned} \tag{2.23}$$

Now, we can refer back to Eq. 2.17, make a few substitutions, and change our notation for transfer entropy to match Schreiber's mutual information definition. Looking at the results, we can conclude that these two concepts are, in fact, equivalent expressions [62, 60].

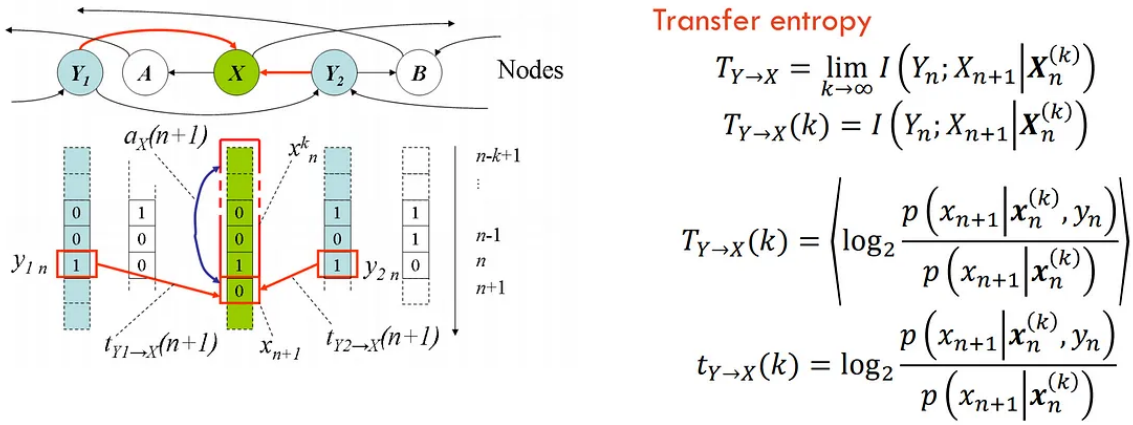


Figure 2.1: Transfer entropy from source Y to target X, with target history length k. Adapted from Joseph Lizier's work on the Java Information Dynamics Toolkit (JIDT) [63].

One important benefit of determining causality with an information-theoretic function such as transfer entropy is that it does not require a particular model to

characterize the interactions between two systems of interest. When contrasted with Granger Causality (GC) or other model-based techniques, transfer entropy's capacity to identify correlations of all orders becomes a significant advantage for exploratory study. This feature is especially useful when trying to determine unknown nonlinear interactions that might be present in the system.

2.2.4 Cross Mappings

Cross-mapping methods operate in state spaces of dynamical systems where shadow manifolds are reconstructed from the lags of the observables. A key feature of these reconstructions is that points that are close in the original state space tend to remain close in the reconstructed space, ensuring that the neighborhoods are preserved. This aspect is crucial, particularly in the context of causality detection.

Cross-mapping methods rely on the topological aspects of the reconstructed attractor dynamics to add a more detailed explanation for how nonlinear interactions cause complex dynamics. Time series variables, according to dynamical systems theory, are casually linked if they share a common attractor manifold [64, 65, 19]. In other words, the process of detecting causality between two systems, X and Y, revolves around examining whether the time indices from the historical data of manifold M_Y can effectively identify proximate points within the reconstructed manifold M_X.

Embedding Theory

The ideas behind the cross-mapping methods are deeply rooted in the theory of time-delay embedding. The initial insight into this concept was provided by Crutchfield [66] and later proven by Taken [65], who also provided a more rigorous and structured framework for the concept. This approach was tested in 1991 [67], then subsequently evolved and generalized, expanding its applicability and scope.[19].

Let ϕ be a dynamic process that defines the temporal evolution of points in a d-

dimensional state space. The paths traced by these points gradually converge towards a manifold M such that the process ϕ acting as a mapping function from M onto itself, symbolized as $\phi : M \to M$. Thus, for any given point $m(t)$ located on the M at time t, its subsequent state at time $t+1$, can be determined by $m(t+1) = \phi(m(t))$ [4].

Consider two time series of length L, $\langle X \rangle = \langle X(1), X(2), \ldots, X(L) \rangle$ and $\langle Y \rangle = \langle Y(1), Y(2), \ldots, Y(L) \rangle$. These time series can be thought of as functions that map points on the manifold M to a sequence of real numbers, i.e., $X(t+1) = X(m(t))$ and $Y(t+1) = Y(m(t))$. The manifolds of X and Y are constructed by using a mathematical property named delay-embedding in nonlinear dynamical systems. This property enables the topological reconstruction of global attractor dynamics based on a sequence of scalar measurements of the variables, which depends on the state of the system [65, 68, 69, 70]. If we have a d-dimensional state space and a positive time lag τ, then we can construct the lagged-coordinate vectors of the shadow manifolds M_X and M_Y as follows:

$$\begin{aligned} x(t) &= \langle X(t), X(t-\tau), X(t-2\tau), \ldots, X(t-(d-1)\tau) \rangle \\ y(t) &= \langle Y(t), Y(t-\tau), Y(t-2\tau), \ldots, Y(t-(d-1)\tau) \rangle \,, \end{aligned} \tag{2.24}$$

where the value of t runs from $1 + (d-1)\tau$ until L.

Generally, the points from the shadow manifolds map 1:1 to points on the original manifold M. However, there are special cases, such as in the canonical Lorenz attractor, where the coordinate Z fails to meet the reconstruction criteria and, hence, does not generate a legitimate shadow manifold. The reason for this failure lies in the symmetrical nature of the two fixed points of the attractor with respect to the Z coordinate. To address this issue and accurately capture the full dynamics of the system, a slight modification, such as a small rotation, can be applied. This adjustment ensures a diffeomorphism (a one-to-one, smooth, and continuously differentiable mapping that preserves the topological structure of the manifold) between the new

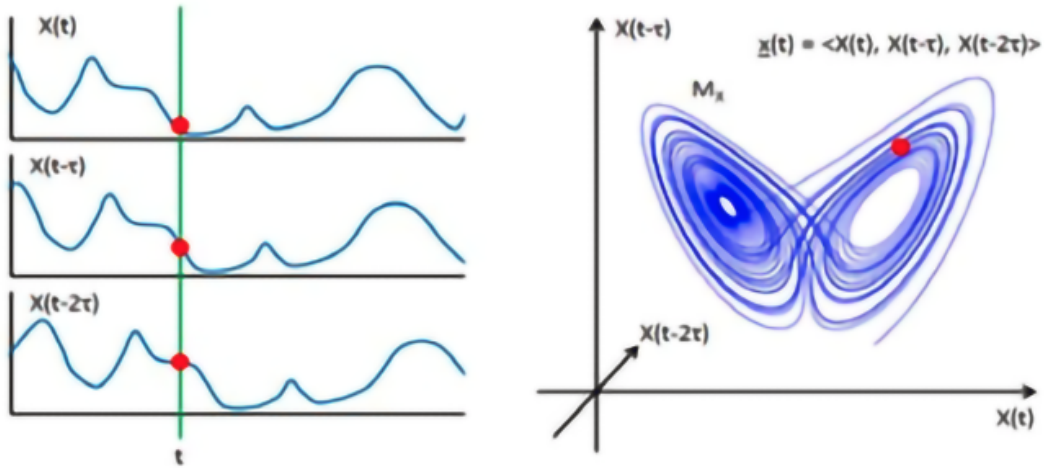


Figure 2.2: Construction of a shadow manifold M_X. Adapted from the work of Luo, Zheng, and Zeng on causal inference in social media using Convergent Cross Mapping [71].

M_Z and M.

Convergent Cross Mapping

Convergent Cross Mapping (CCM) was introduced in [4] to determine causation in real systems. This method is based on simplex projection [72], a robust technique that relies on exponentially weighted distances from points in close proximity on a reconstructed manifold to do kernel density estimation [65].

Consider the time series of length L and their lagged-coordinated vectors discussed in section 2.2.4. We begin by finding the $d+1$ nearest neighbors of the lagged-coordinate vectors on the shadow manifold M_X to generate a cross-mapped estimate of $Y(t)$. Next, we find the time indices from closest to farthest corresponding to the $d+1$ nearest neighbors of $x(t)$ on M_X. These time indices are used to identify the analogous points (neighbors) in Y to estimate $Y(t)$ from a weighted mean of the $Y(t_i)$

values, where $i = 1, \ldots, d+1$.

$$\hat{Y}(t)|M_X = \sum w_i Y(t_i), \tag{2.25}$$

where the weighting function w_i is based on the distance between $x(t)$ and its i^{th} nearest neighbor on M_X. Thus, the weights can be determined by the following equations

$$w_i = u_i / \sum_{j=1}^{d+1} u_j, \tag{2.26}$$

where

$$u_i = e^{-\frac{d[x(t),x(t_i)]}{d[x(t),x(t_1)]}}, \tag{2.27}$$

and $d(X, Y)$ is the Euclidean distance between vectors X and Y. Analogously, we can define the CCM from Y to X

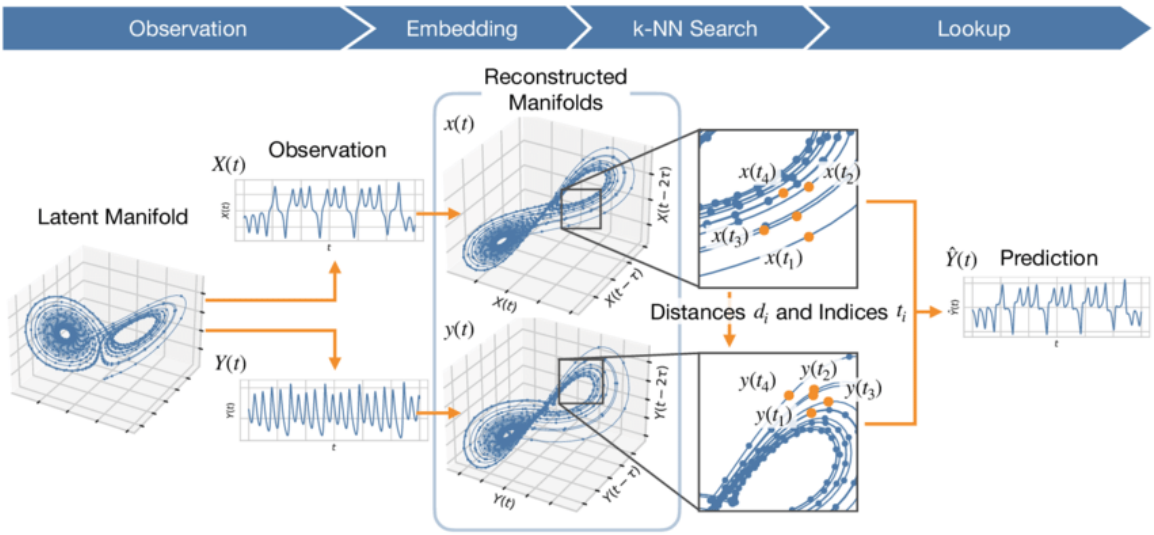


Figure 2.3: Overview of CCM. Adapted from the work by Takahashi *et al.* on performance-portable implementation of Empirical Dynamic Modeling using Kokkos [73].

The causal discovery from X to Y is derived from the correlation coefficient ρ between Y and $\hat{Y}(t)|M_X$. By analogy, if Y drives X, a high value of correlation between X and $\hat{X}(t)|M_Y$ is expected. Low values of correlation are indications of a

poor or complete absence of a causal link between variables of the system.

In their study, the researchers focused on observing the convergence of cross-mapped estimates toward accurate values as the number of data points used increased. This approach is particularly effective in distinguishing causally coupled systems from those that are merely correlated. The amount of data required for an accurate reconstruction is dependent on the dimensionality of the attractor. Thus, in systems where there is a causal connection, the accuracy of the estimates tends to improve as the length of the time series increases, demonstrating a direct relationship between the quantity of the data and the reliability of the causal inference.

State space reconstruction methods are most effective in scenarios where the system under study is nonlinear and can be represented in a relatively low number of dimensions. Additionally, these methods yield better results when the observational noise within the system is manageable.

2.2.5 Causal Neural Networks

Neural networks excel in multiple fields, including computer vision and natural language processing, among other things. Specifically, Convolutional Neural Networks (CNNs) have drastically changed areas like computer vision and pattern recognition, where a grasp of spatial hierarchies is critical. However, these traditional networks are fundamentally "associative," which means they are great at finding correlations but not necessarily causations. In order to close this knowledge gap, causal networks have arisen with the goal of better comprehending causal interactions in both time and spatial domains. Before exploring some methods that employ CNNs, let's first clarify a few key definitions.

Convolutional Neural Networks: A convolutional neural network, also known as CNN or ConvNet, is a sophisticated type of neural network that is highly effective

in handling data represented in a grid format, such as images, videos, and even multi-dimensional time-series data. This specialized network architecture has several unique features and improvements over traditional feed-forward neural networks. Conventional machine learning techniques require hand-crafted extraction and selection of features. CNNs, on the other hand, can perform feature engineering automatically by identifying these features independently through training. Moreover, common issues, such as vanishing and exploding gradients, are prevented by employing regularized weights across a reduced number of connections during the backpropagation process [74].

A CNN is primarily composed of three types of layers: convolutional layers, pooling layers, and fully connected layers. The convolution and pooling layers extract features, while the fully connected layer transforms these features into final outcomes, like classifications.

The convolution layer is essential to the CNN's structure, carrying most of the computational load. This layer employs three key concepts that are crucial to its operation [75]. First, convolutional layers employ sparse interaction, which involves employing fewer weights (or filters) to interact with specific portions of the input and concentrate on localized regions. Second, weights are shared across various input components through parameter sharing. The same filter (set of weights) is used across the entire input, allowing for the detection of a feature regardless of its position in the input. Third, equivariant representations guarantee that if the input changes (e.g., shifts), the output changes in the same manner.

Pooling layers are used after convolutional layers in order to minimize the spatial dimensions (width and height) of the input volume. This reduction in dimensionality means that the network requires less memory and has less computing burden because the following layers have fewer parameters. Pooling layers, despite reducing the size of the input, preserve the most important information of the network. They accomplish

this by summarizing (max or average) the presence of features in patches of the feature map. Pooling helps the network achieve translation invariance. In other words, once a feature is identified, its precise location loses significance.

Fully connected layers are positioned towards the end of the network. They handle all the high-level reasoning in the network after all the convolutional and pooling layers. These layers are where the network makes decisions based on the features extracted and condensed by previous layers. They can be considered a part of the network's decision-making process.

Causal Convolutional Networks: Causal Convolutional Networks are a specialized type of neural network that integrates causal inference into convolutional frameworks. One important consideration in many convolutional applications, especially audio processing applications, is maintaining the temporal order of the input signal. The time-domain signal in these applications is a series of samples that need to be handled with consideration for the temporal order of the sequence. According to this criterion of causal processing, output at any given point in time should only depend on input samples that came before it, not on samples that will come after.

Standard convolutional operations do not inherently meet this causal requirement. They take into account all input samples that fall within the filter's coverage, including those from the future. This means that the convolution at each time step incorporates future information, thereby violating the principle of causality.

Causal convolutions are designed to address this issue by restricting the input that may be accessed at each time step to only include data up to the current timestep. This method guarantees that the output at any given time is solely impacted by past and present data. One common approach is to add padding to the input data in such a way that the principle of causality is always satisfied by making the convolutional filters access only current and past data. This technique is known as "masked

convolution" or "causal padding."

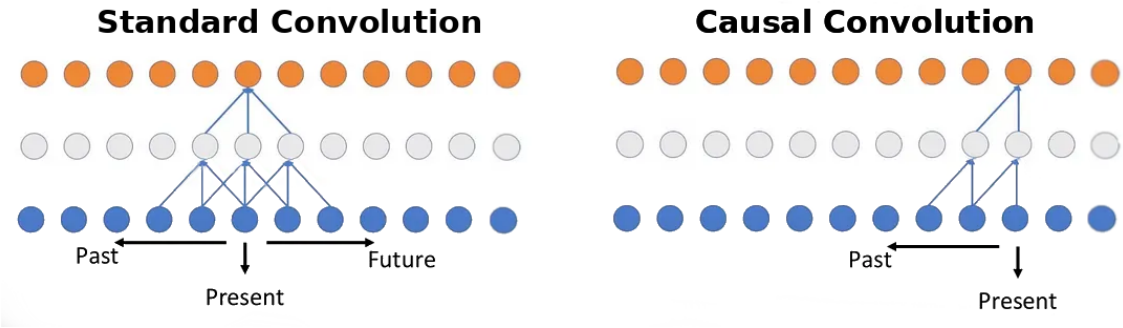


Figure 2.4: Standard vs Causal Convolution.

Unlike standard convolutional layers that use a fixed kernel, causal convolutions employ a kernel that adapts based on the current time step. This is achieved by creating a kernel mask, a binary tensor that matches the dimensions of the convolutional kernel. The mask has zeros in positions corresponding to future values, which essentially prevents them from having an impact on the current output. By applying the causal convolution in conjunction with this masked kernel, the network makes predictions based only on past and present data. The temporal order is preserved as this technique successfully stops any data leakage from future time steps into the model.

2.2.6 Neural Granger Causality

In order to detect Granger causality in nonlinear settings, Tank *et al.* introduced the Neural Granger method [76]. This method includes the use of either structured multilayer perceptrons (MLPs) or recurrent neural networks (RNNs), along with penalties that induce sparsity in their weights. By utilizing convex group-lasso penalties, this strategy concentrates on pushing specific groups of weights to zero. Unlike conventional techniques, this framework can effectively identify long-range connections between series.

Consider x_t as a p-dimensional stationary time series within the $\mathbb{R}^p$ space, observed

across a span of T time points, denoted as $(x_1, \ldots, x_T)$. In the realm of nonlinear autoregressive modeling, the trajectory of x_t unfolds according to more general nonlinear dynamics

$$x_t = g(x_{<t1}, \ldots, x_{<tp}) + e_t, \tag{2.28}$$

where

$$x_{<t1} = \left(\ldots, X_{(t-2)i}, x_{(t-1)i}\right). \tag{2.29}$$

Here, $x_{<t1}$ captures the past of the i series. Within this framework, the function g operates as a nonlinear autoregressive function, which can be dissected into distinct components corresponding to each series. For any given series i, this relationship is represented as:

$$x_{ti} = g_i(x_{<t1}, \ldots, x_{<tp}) + e_{ti}. \tag{2.30}$$

Here, g_i maps out the influence of the past K lags on the current state of series i, thereby linking historical data to present values.

In this context, Granger non-causality, especially between two series j and i, hinges on the independence of g_i from $x_{<tj}$, the historical lags of series j. This suggests that the historical values of series j do not contribute to or influence the future trajectory of series i as determined by g_i.

Sparse Input MLPs for Time Series: A Multilayer Perceptron (MLP) is a type of modern feed-forward artificial neural network, i.e., information moves in only one direction—from input nodes, through hidden layers, to output nodes—without any loops or cycles. It is widely used in machine learning for various tasks, including classification, regression, and pattern recognition.

An MLP consists of fully connected neurons with nonlinear activation functions that are organized in three or more layers, including an input layer, one or more hidden layers, and an output layer. It is notable for being able to distinguish data

that is not linearly separable.

Often referred to as "vanilla" neural networks, MLPs are trained by a technique called backpropagation [77]. The procedure involves calculating the error that exists between the network's output and the actual target values. This error is then propagated back through the network to modify the weights and biases. This is often done using gradient descent or variations.

Neural Granger uses a distinct MLP to model each component g_i, which we refer to as a componentwise MLP (cMLP). This technique simplifies the process of distinguishing the influences of inputs on the outputs. For every component i, the function g_i is configured as an MLP with L layers. We denote the values in the l-th hidden layer at time t as h_t^l. The weights across the layers are represented as $W = \{W^1, \ldots, W^L\}$, with the initial layer weights specified as $W^1 = \{W^{11}, \ldots, W^{1K}\}$. The hidden values at the first layer at time t are calculated by:

$$h_t^1 = \sigma\left(\sum_{k=1}^{K} W^{1k} x_{t-k} + b^1\right), \tag{2.31}$$

where σ is the activation function, and b_1 is the bias at the first layer. Subsequent layers are composed of fully connected units with σ activation functions. The output x_{ti} is then given by:

$$x_{ti} = g_i(x_{<t}) + e_{ti} = w_O^T h_t^L + e_{ti}, \tag{2.32}$$

where w_O^T is the linear output decoder, and h_t^L is the final hidden output from the L-th layer. This structure enables each g_i to effectively model complex relationships within the data, leveraging the depth and nonlinear capabilities of MLPs.

It may be inferred that the time series j does not Granger cause series i if, for all of k, the j-th column of the first layer weight matrix $W_{:j}^{1k}$ in Eq. 2.31 is made up entirely of zeros. Thus, analogously to the VAR case, with a group lasso penalty applied to the columns of the W^{1k} matrices for each g_i, it is possible to select for

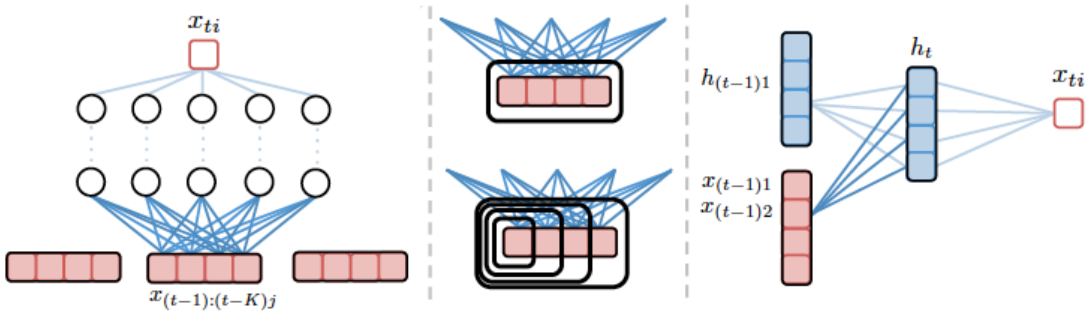


Figure 2.5: Neural Granger using cMLPs. Adapted from the research on Neural Granger Causality by Tank *et al.* [76].

Granger causality, as indicated by the following equation:

$$\min_{W} \sum_{t=K}^{T} \left(x_{it} - g_i(x_{(t-1):(t-K)})\right)^2 + \lambda \sum_{j=1}^{p} \left\|(W_{:j}^{11}, \ldots, W_{:j}^{1K})\right\|_F, \tag{2.33}$$

where λ is a regularization parameter. For a sufficiently large λ, the solutions to this equation will lead to many zero columns in each W^{1k} matrix, suggesting that only a few Granger causal connections are estimated. This method effectively identifies significant relationships in the data by penalizing and thus simplifying the complexity of the model.

Sparse Input Recurrent Neural Networks: Recurrent Neural Networks (RNNs) represent a sophisticated class of artificial neural networks specifically engineered for discerning patterns in sequential data types, including but not limited to textual content, genomic sequences, handwriting styles, and spoken language (more detailed information can be found in Sec. 2.3.3). Unlike traditional neural networks, which operate under the assumption that inputs and outputs are mutually exclusive and unrelated, RNNs are particularly well suited to modeling time series data. This proficiency comes from their ability to effectively compress historical data from a time series into a distinct hidden state, allowing them to unravel and understand complex

temporal patterns and nonlinear relationships that extend over longer periods than what is typically achievable with standard time series models.

We take a similar approach to MLPs in order to capture Granger causality using RNNs, where each function g_i is modeled using a single RNN. Let $h_{t-1} \in \mathbb{R}^m$ denote the m-dimensional hidden state at time t, which captures the historical background of the time series necessary for forecasting the component x_{ti}. This hidden state at time $t+1$ is recursively updated as:

$$h_t = f(x_t, h_{t-1}). \tag{2.34}$$

In this equation, f represents some nonlinear function that depends on the particular recurrent architecture. To model this function, we utilize Long Short-Term Memory (LSTM) networks due to their proven capability to handle complex temporal dependencies (more detailed information can be found in Sec. 2.3.3). LSTMs are particularly adept at maintaining information over extended periods, which is essential for accurately modeling the time-dependent relationships inherent in Granger causality analysis.

To simplify this approach, the output $g_i(x_{<t})$ is expressed as a linear function of the hidden states at time t:

$$x_{ti} = g_i(x_{<t}) + e_{ti} = w_O^T h_t + e_{ti}. \tag{2.35}$$

In the LSTM framework, we introduce an additional hidden state variable c_t, known as the cell state, leading to the complete set of hidden parameters being

(c_t, h_t). The standard LSTM model is formulated as follows:

$$\begin{aligned} f_t &= \sigma(W^f x_t + U^f h_{t-1}), \\ i_t &= \sigma(W^{in} x_t + U^{in} h_{t-1}), \\ o_t &= \sigma(W^o x_t + U^o h_{t-1}), \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W^c x_t + U^c h_{t-1}), \\ h_t &= o_t \odot \tanh(c_t), \end{aligned} \tag{2.36}$$

where $\odot$ represents element-wise multiplication. The terms i_t, f_t, and o_t correspond to the input, forget, and output gates, respectively. These gates regulate the updates to each component of the cell state c_t and its transfer to the hidden state h_t, which is used for predictions. The sigmoid function σ and the hyperbolic tangent function tanh are applied element-wise, allowing the LSTM to effectively manage the flow of information and update its memory (cell state) for accurate predictive modeling.

The set of input matrices we used in Eq. 2.36 form the block matrix W, represented as:

$$W = \left((W^f)^T, (W^{in})^T, (W^o)^T, (W^c)^T\right)^T, \tag{2.37}$$

where W governs how the previous time series data, x_t, affects the various gates of the LSTM –namely, the forget, input, and output gates– as well as the updates to the cell state. Consequently, it influences the evolution of the hidden state. In this componentwise LSTM model (cLSTM), similarly to the MLP methodology, a sufficient condition for Granger non-causality from an input series j to an output series i is indicated if all the elements of the j-th column of W are zeroes. Therefore, to determine which series Granger causes series i during the model's estimation phase,

a group lasso penalty is applied to the columns of W:

$$\min_{W,U,w^O} \sum_{t=2}^{T} (x_{it} - g_i(x_{<t}))^2 + \lambda \sum_{j=1}^{p} \|W_{:j}\|_2, \tag{2.38}$$

where $W_{:j}$ represent all the elements of the j column. Here, U is defined as:

$$U = \left((U^f)^T, (U^{in})^T, (U^o)^T, (U^c)^T\right)^T. \tag{2.39}$$

Setting a high value for λ leads to many zero columns in W, effectively creating a model of sparse Granger causal links.

2.2.7 TCN Introduction

Temporal Convolutional Networks (TCN) were first introduced by Lea *et al.* [78] for video-based action segmentation. They have been specifically adapted to handle one-dimensional temporal data, making them highly effective for analyzing time series. Their rise in popularity can be attributed to their ability to extend most of the convolutional advantages of regular CNNs, such as sparsity and translational equivariance, into the time domain. Recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) have historically dominated the field of sequential data processing. However, due to their lack of effectiveness, TCNs were created in response to the necessity for more efficient methods. TCNs provide a flexible and efficient framework for most sequence modeling tasks, with fewer parameters and fewer memory requirements.

TCNs provide a reliable and effective alternative for sequence modeling, particularly in scenarios where it is crucial to understand the long-range temporal dependencies of the data. They are an important tool in the fields of data analysis and machine learning because of their versatility, capacity for parallel processing, and ability to

handle long sequences. TCNs outperform recurrent neural networks (RNNs) and long short-term memory (LSTM) models when dealing with long sequences as they can overcome issues like vanishing gradients commonly encountered in these traditional models. However, effective training of TCNs often requires a substantial amount of data to accurately capture intricate temporal patterns. Additionally, TCNs are sensitive to hyperparameter selection, including the size of the convolutional filters and the dilation rates.

TCN Architecture

A TCN is formed by several causal, dilated 1D convolutional layers with the same input and output lengths able to encode the spatial-temporal information of the time series. A key characteristic of this framework is its simplicity, long memory, and ability to outperform most convolutional architectures during auto-regressive prediction tasks [79].

Dilated causal convolutions are the most important component of TCNs. The dilated causal convolution that is used in the TCN architecture is derived from the WaveNet paper [80]. The number of time steps seen by the sliding kernel, also known as the receptive field, is equal in a single-layer TCN to the size of the kernel specified by the user. The receptive field in TCNs adapts based on the dilation rate and the depth of the network. This adaptability allows the network to focus on learning long-range dependencies without increasing the computational complexity. In order to uncover the causal relationship, the receptive field has to be no less than the delay between cause and effect. By "causal," we understand that there is no information leakage from the future to the past, e.g., a filter can only use inputs at time step t and earlier.

A dilated convolution skips input values with a certain step size, called dilation factor, by applying a kernel over an area larger than its size [81]. This dilation

factor can increase exponentially, which facilitates a network with stacked dilated convolutions to work on a large area without loss of resolution or coverage. This results in an exponentially larger receptive field containing fewer parameters and layers.

Another component of TCNs is residual blocks, which originated from ResNet [82]. Two dilated causal convolution layers are stacked together, and the results from the last convolution step are then added to the inputs to extract the output from the block, thus enabling a long, effective history. In the case where the number of channels of the inputs and the number of filters are different, a 1D convolution is applied to the inputs.

Let's delve deeper into some of these concepts to better understand their applications and implications.

Dilated Convolutions: Dilated convolutions are methods utilized in CNNs that are particularly beneficial for tasks that require the capture of long-range context without sacrificing resolution. The kernel (or filter) is expanded by inserting gaps (or "dilations") between its consecutive elements. This expansion of the receptive field allows the filter to cover a larger area of the input feature map without increasing the size of the filters and facilitating the learning of more intricate representations of the input data.

Dilated convolutions differ from standard convolutions by adding holes within the kernel elements, which are controlled by the dilation rate. This hyperparameter controls the distance between the filter elements. Dilated convolution is identical to regular convolution when the dilation rate is set to 1. One can visualize this as applying an array of filters across the input data, with each filter sampling one data point but spaced apart according to the dilation rate. Greater dilation rates widen the receptive field by increasing the distance between the filter's elements.

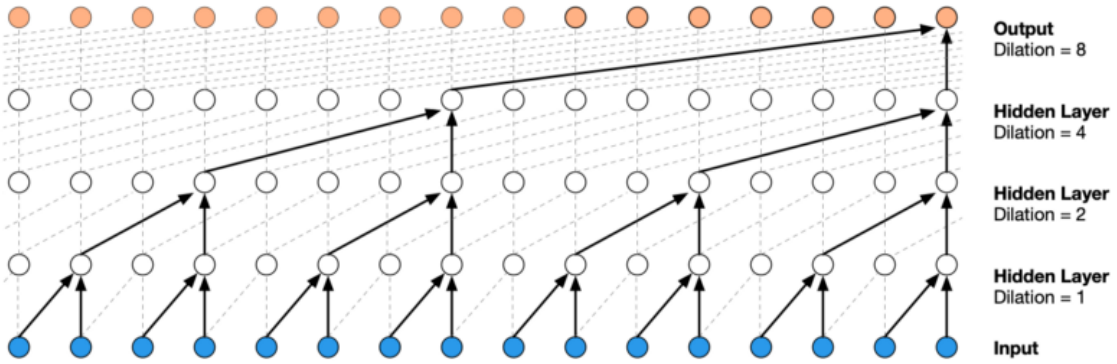


Figure 2.6: Stack of dilated causal convolutional layers. Adapted from the work by van den Oord *et al.* on WaveNet, a generative model for raw audio [80].

Residual Connections: Residual Connections, also known as skip connections, offer a novel approach to connecting layers in a neural network. They were designated to make training very deep neural networks easier. As networks become more complex, residual connections prevent the gradients from becoming too small for effective learning during backpropagation.

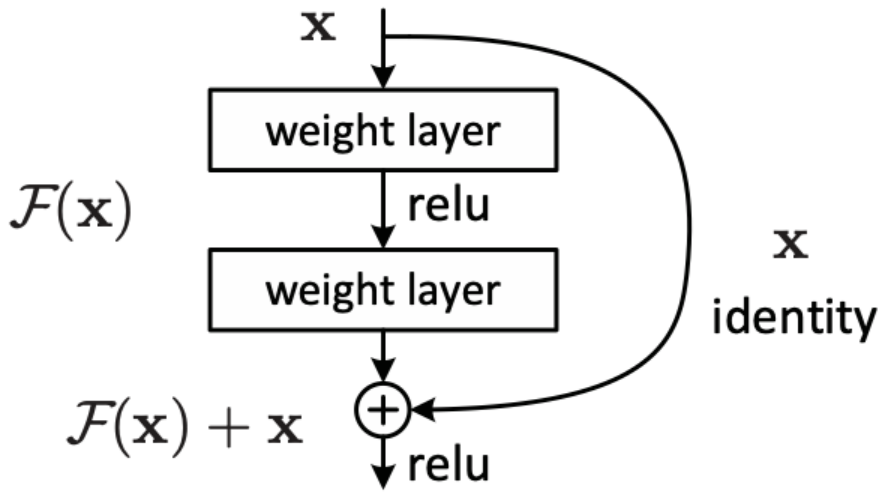


Figure 2.7: Residual Connections. Adapted from He *et al.*'s foundational study on deep residual learning for image recognition [82].

In a typical layer of a neural network, the input is transformed through weights, biases, and activation functions to produce an output. In a network with residual

connections, a shortcut or "skip" connection is created, allowing the output of an earlier convolutional layer to effectively skip a number of intermediate convolutional steps.

In its most basic form, these connections perform an identity mapping, in which the input is added directly to the output of the residual block. This prevents the degradation issue that might arise in very deep networks and guarantees that the deeper levels can perform as well as the shallower layers. Moreover, deeper layers can focus on learning the residuals instead of learning the complete representation from scratch. These connections improve the flow of gradients during backpropagation, making it easier to train deeper models.

Autoencoders: Kramer initially introduced the autoencoder as a nonlinear extension of principal components analysis (PCA) [83]. An autoencoder represents a form of artificial neural network designed for unsupervised learning, with the objective of efficiently discovering representations of unlabeled data. It has two primary functions: the first is to encode the input data into a new representation, and the second is to decode the encoded information in order to recreate the original input.

The primary purpose of an autoencoder is to discover a compressed and efficient encoding of a dataset. They are typically used for the purpose of dimensionality reduction or feature learning, although over time, their application has expanded so that they are now widely used for learning generative models of data. The structure of an autoencoder consists of three key components:

- *Encoder:* This part of the autoencoder is a feed-forward, fully connected neural network consisting of a sequence of convolutional blocks followed by pooling modules. Each convolutional block in the encoder applies a series of filters to successfully extract and highlight different features and patterns of the input data. Following this process, these features are passed through pooling modules,

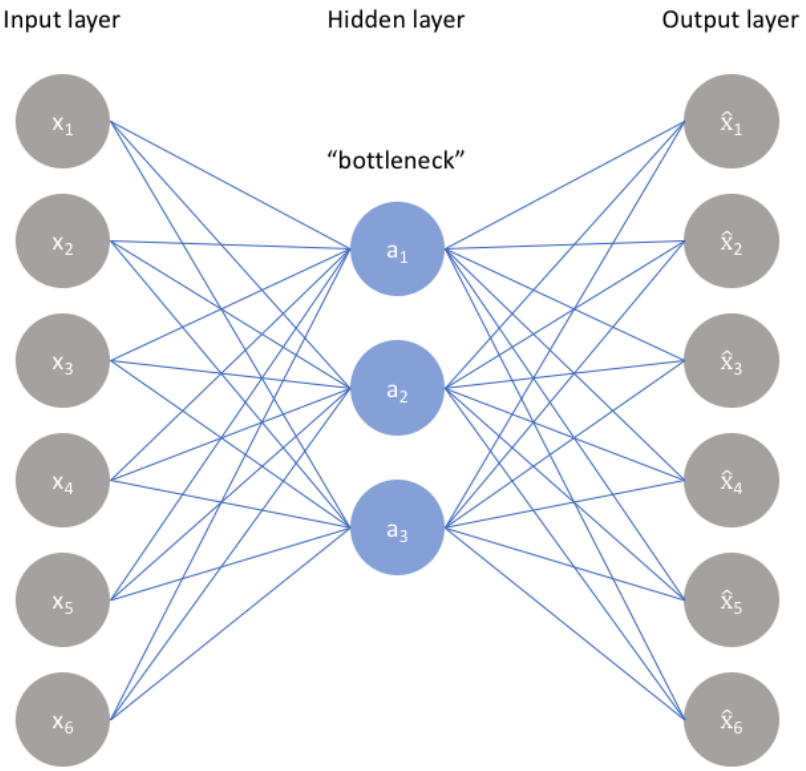


Figure 2.8: Autoencoder architecture. Adapted from Jeremy Jordan's exploration of autoencoders [84].

which help to decrease the spatial dimensions of the data. Convolution and pooling progressively compress the data while extracting the most important information at the same time. The encoder learns to preserve as much of the significant information from the input data as possible but in a more compact form. This representation is typically several orders of magnitude smaller than the input data.

- *Bottleneck:* Also known as the latent space or hidden layer, the bottleneck is the most important part of the neural network. It regulates the flow of information from the encoder to the decoder, ensuring that only the most important elements of the input data are allowed to pass through. The bottleneck also prevents the neural network from simply memorizing the input data. This is crucial for time series analysis, where discovering trends and patterns is more important than just memorization. However, the size of the bottleneck must be chosen carefully. A well-calibrated bottleneck allows the network to learn and interpret these complex patterns present in time series data. On the other hand, if the size is too small, it might be more detrimental than beneficial. An excessively restricted bottleneck might prevent the network from capturing all the necessary data, which increases the chances of important information getting leaked through the pooling layers. Therefore, it is vital when designing an autoencoder to find the proper size so that we can use them to their full potential. It should be big enough to capture the important temporal dynamics and patterns in the data but not so big that it makes the network prone to overfitting.

- *Decoder:* The decoder has a similar structure to the encoder but in reverse order. It consists of a series of upsampling and convolutional blocks, each meticulously designed to reconstruct the output originating from the bottleneck. This step is essential in order to recover the spatial resolution that was compressed dur-

ing the encoding process. After upsampling, the convolutional blocks are used to refine the upsampled data. The finer details and structures of the original input are gradually restored by the use of filters that can extract and enhance its elements. This process involves not just a merely straightforward data augmentation but a careful and precise decompression from its latent attributes, ensuring that the reconstructed time series faithfully mimics the original data.

An autoencoder's training begins with a forward pass, where the input data is first compressed by the encoder and then reconstructed by the decoder. Once that is done, the autoencoder calculates the loss by measuring the difference between the original input and its reconstruction. This metric is extremely important since it quantifies the autoencoder's proficiency in capturing and reconstructing the input data. Generally, we use Mean Squared Error (MSE) for continuous data and Binary Cross-Entropy for binary data.

Backpropagation is the primary method by which the autoencoder learns the underlying pattern and structure of the input data. During this process, the gradient of the loss function is calculated with respect to each weight in the network. The gradients are calculated in reverse order, starting from the output and then traveling back through the network. The idea behind this step is to identify how each weight contributes to the error and what adjustments are needed to reduce it.

After calculating the gradients, the neural network adjusts its weights and biases using an optimization algorithm like Stochastic Gradient Descent (SGD), Adam, or RMSprop. The learning rate is one of the most important hyperparameters, and it determines the size of the steps necessary to minimize the loss. It's crucial to find the right balance: a learning rate that is too high can overshoot the minimum, and a rate that is too low can either slow down the convergence or get stuck in local minima.

The entire process of forward pass, loss computation, backpropagation, and weight updating is repeated across multiple iterations, known as epochs. During each epoch,

the entire dataset is passed through the autoencoder. This process is repeated until the loss stops changing significantly, implying that the autoencoder has successfully learned to represent the input data accurately.

2.2.8 Temporal Causal Discovery Framework

The Temporal Causal Discovery Framework (TCDF) developed by Nauta *et al.* [81] represents an innovative approach in the field of causal discovery, particularly for analyzing time series data. TCDF aims to construct a temporal causal graph that reveals the causal relationships and time delays between various observed continuous time series within a dataset. They do this by utilizing attention-based Convolutional Neural Networks (CNNs) in conjunction with a causal validation step. TCDF also tries to solve the typical challenges encountered by several previous methods when facing complex causal models.

Figure 2.9: Examples of causality confounds. Left: X_1 is the direct cause of X_2 with a one-time step delay and influences X_3 indirectly, resulting in a cumulative delay of 4-time steps $(1 + 3)$. Right: X_1 acts as a common cause for X_2 and X_3, influencing them with delays of 1 and 4 time steps, respectively. Adapted from Nauta *et al.*'s study on causal discovery with attention-based convolutional neural networks [81].

First, it is crucial to identify whether a causal relationship is direct (i.e., one variable directly influences another) or indirect (i.e., the influence is mediated through one or more intermediate variables), as we can see in Figure 2.9 left. TCDF utilizes the attention mechanism within CNNs to focus on the specific inputs (time series) that are most predictive of a target series. This focus helps to infer direct causality. The causal validation step further refines these relationships by assessing the impact

of potential causes on the target series, thus distinguishing between direct and indirect causes. Second, some causal effects occur almost instantaneously, meaning there is little to no time delay between the cause and its effect. Detecting these effects can be difficult, especially in time series data where the resolution is not good enough to capture rapid changes. Third, accounting for confounders that could potentially introduce spurious correlations and incorrect causal inferences. We can see in Fig 2.9 right how the presence of X_1 influencing X_2 and X_3 could mislead causal methods into assuming a causal link between X_2 and X_3.

The Attention Mechanism: This methodology builds upon the foundational Temporal Convolutional Network (TCN) model (more about this topic in Section 2.2.7). The standard TCN model, due to its univariate nature, restricts its application to multivariate time series analysis. In a standard deep TCN setup, the addition of 1D-convolutional layers to the architecture results in a one-dimensional output from each convolutional layer. Such a configuration inherently mixes the input signals, which can prevent the network from learning an accurate causal discovery.

The univariate TCN architecture can be extended into a one-dimensional depthwise separable convolutional architecture to better accommodate the challenges of multivariate time series analysis. This modification employs depthwise convolutions to independently apply a unique kernel to each input channel, ensuring that each input time series is processed in isolation. This step is followed by a pointwise convolution of size 1x1 that merges the outputs from all channels into a unified representation. This adjustment diverges from traditional convolutional network designs, which typically use a single kernel across all layers to process combined inputs.

In practical terms, the TCDF architecture consists of N distinct channels, each dedicated to one of the input time series. Each of these channels is formed by an attention-based Convolutional Neural Network (CNN). The architecture of TCDF,

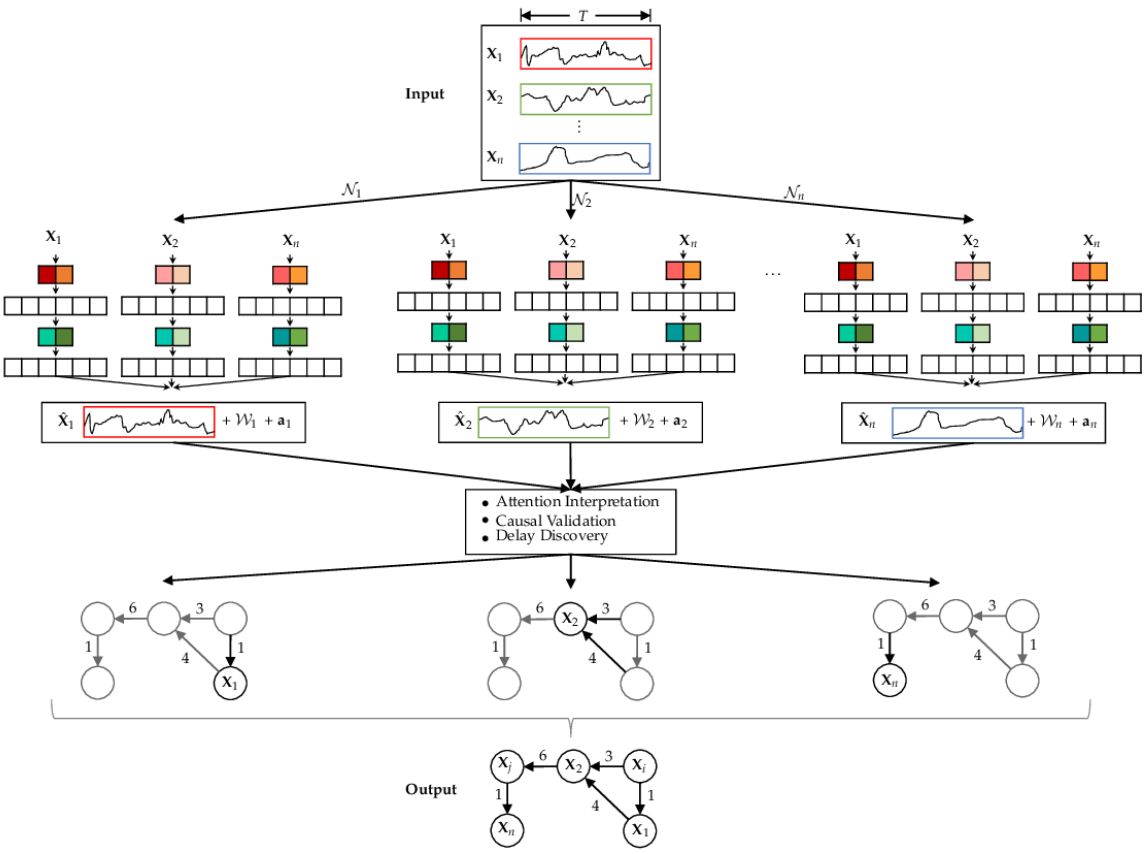


Figure 2.10: The Temporal Causal Discovery Framework (TCDF) employs N independent Convolutional Neural Networks (CNNs), denoted as $N_1, N_2, \ldots, N_n$, where each network is tasked with processing time series inputs $X_1, X_2, \ldots, X_n$ of length T. Each network N_j is designed not only to forecast the future values $\hat{X}_j$ of its corresponding time series X_j but also to produce the associated kernel weights W_j and attention scores a_j. Adapted from Nauta *et al.*'s study on causal discovery with attention-based convolutional neural networks [81].

illustrated in Figure 2.10, shows how each N_j channel is optimized to forecast the values of its corresponding target series X_j. The objective of each network N_j is to minimize the loss L between the actual observed values of X_j and its predictions $\hat{X}_j$. The input for any given channel N_j is the dataset X, which is composed of N time series, each extending over T time steps. Row X_j corresponds to the target series, denotes as $X_j = \left[0, X_j^1, X_j^2, \ldots, X_j^{T-1}\right]$, while all other rows are assigned to the exogenous time series $X_{i \neq j} = \left[0, X_i^1, X_i^2, \ldots, X_i^{T-1}\right]$

The network's architecture can be further enhanced by incorporating an attention mechanism. This mechanism tells the network where to focus when predicting a time series. Attention is implemented through a trainable vector a of dimension $1 \times N$, where N is the number of input time series. Each element within this vector is known as an attention score. For every network N_j, a unique attention vector $a_j = [a_{1,j}, a_{2,j}, \ldots, a_{i,j}, \ldots, a_{N,j}]$. The attention score $a_{i,j}$ is specifically assigned to input time series X_i in network N_j. This indicates the degree of focus N_j places on X_i for the prediction target X_j. A higher score for $a_{i,j}$ suggests a possible causal relationship where X_i influences X_j, while a lower score implies a lesser or no causal link.

The attention mechanism is initially implemented through a soft attention strategy, employing the Softmax function s to each element a within a_j during each epoch of training. After the training of network N_j is finished, a semi-binarization function, dubbed HardSoftmax, which zeros out any attention scores not meeting a specified threshold τ_j, is applied.

$$h = \text{HardSoftmax}(a) = \begin{cases} \sigma(a) & \text{if } a \geq \tau_j, \\ 0 & \text{if } a < \tau_j. \end{cases} \tag{2.40}$$

Here, h_j is the subset of attention scores in a_j subjected to the HardSoftmax function. A time series X_i is considered a potential influencer of the target series X_j if its

corresponding adjusted attention score $h_{i,j}$ exceeds 0. This approach allows TCDF to systematically identify and enumerate potential causal relationships across the dataset's time series, encouraging a better understanding of the underlying causal structures.

2.3 Methods for Behavioral States Discovery

2.3.1 Dynamical systems

The field of dynamical systems theory is built upon the principles of motion and forces that govern physical systems. This foundation stretches back to the early formulations of classical mechanics. It provides a comprehensive framework for understanding the evolution of various systems over time and extends its use from simple mechanical structures to complex biological networks. A dynamical system in an n-dimensional space $\mathbb{R}^n$ is defined by n first-order differential equations, which model the temporal dynamics of evolutionary processes. The 'state' of such a system includes a wide range of variables that are crucial to the specific context under study. For example, in the field of ecology, it could refer to the fluctuating populations of predators and prey.

Dynamical systems are categorized into types such as linear versus nonlinear and deterministic versus stochastic. Linear systems are usually easier to analyze and forecast due to the straightforward proportionality between inputs and outputs. On the other hand, nonlinear systems are harder to analyze due to unexpected and complex behaviors. For a continuous-time dynamical system, the representation can be given as:

$$x = x(t) \in \mathbb{R}^n, \quad t \in I \subseteq \mathbb{R} \tag{2.41}$$

$$\frac{dx}{dt} = \dot{x} = f(x, t), \tag{2.42}$$

where x represents the system's dynamics, and $f(x,t)$ is a smooth function defined in a subset $U \subseteq \mathbb{R}^n \times \mathbb{R}$. In this model, t usually represents time, and $f(x,t)$ is typically nonlinear [85].

An important element in the analysis of dynamical systems is the concept of phase space. It includes all possible states of the system. By examining the trajectories within this phase space, we can discover the temporal behavior of the system. Moreover, the study of dynamical systems involves an understanding of how these systems adapt and respond to varying environmental conditions or parameter changes.

Basins of Attraction

The idea of basins of attraction is central to the framework of dynamical systems. It explains how initial states evolve over time toward specific outcomes or attractors. These basins include the entire range of possible starting conditions that influence the long-term behavior of the system and ultimately guide it toward a specific attractor. The system can display a range of behaviors, including quasiperiodic patterns, chaotic dynamics, and periodic oscillations, depending on the type of attractor. If there is a single, global attractor, the analysis is simpler since every initial condition within the state space will eventually converge to this unique attractor over time. However, many dynamical systems display multistability, where multiple attractors exist simultaneously, each with its own basin of attraction [86]. Therefore, because of this phenomenon, the state space is divided into distinct regions, each of which directs the initial conditions towards its corresponding attractor.

The complexity of basins of attraction often makes typical analytical approaches inadequate for their study. Instead, numerical simulations become the primary tool for exploring these basins. Therefore, researchers need to define the basins of attraction and carefully trace the trajectories from a wide array of initial conditions to their eventual attractors.

Attractors themselves can take various forms, including stable points (equilibria), limit cycles (periodic orbits), attracting torii (quasiperiodic orbits), or strange attractors (chaotic behavior). The basin of attraction surrounding each type of attractor can vary greatly in size and shape, from narrowly encircling the attractor to encompassing the entire state space. Most commonly, a basin of attraction occupies a sizeable but finite portion of the state space, potentially extending infinitely in certain directions while remaining bounded in others [87]. This spatial diversity highlights how the complicated interaction between initial conditions and system dynamics impacts the evolution and ultimate behavior of these systems.

2.3.2 Fixed Points

A fixed point in the context of dynamical systems is a specific state or value within the phase space where the system's state remains constant over time. When a system reaches a fixed point, it will stay in this state indefinitely, assuming there are no external influences to perturb it. Mathematically, for a system represented by a function f, a fixed point x^* is defined such that $f(x^*) = x^*$ for discrete systems. For continuous systems, this condition is represented by $\dot{x} = 0$, where $\dot{x}$ denotes the time derivative of x, indicating a state where there is no change in the system's state over time.

Fixed points can exhibit different stability characteristics: stable, asymptotically stable, or unstable [88]. A stable fixed point means that trajectories that start near the fixed point will remain close to it over time. An asymptotically stable fixed point not only maintains the closeness of trajectories but also ensures that these trajectories will converge towards the fixed point as time progresses towards infinity. On the other hand, an unstable fixed point acts repulsively, causing trajectories that start near it to diverge away as time moves forward. In complex dynamical systems, determining and assessing the stability of fixed points is an essential but difficult task. However,

in many systems, such as harmonic oscillators, pendulums, and fluid flows, the trivial solution of $u^* = 0$ is a fixed point.

Fixed points, in the domain of ordinary differential equations (ODEs), are constant solutions that are necessary to comprehend the system's behavior. Therefore, from a stability standpoint, we can better understand the asymptotic properties of solutions and trajectories close to these fixed points. Similarly, in systems governed by partial differential equations (PDEs), fixed points facilitate the pursuit of steady-state solutions, which are vital due to their ability to represent systems in equilibrium.

2.3.3 Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are a class of artificial neural networks that are especially important in the studies of animal behavior because of their ability to recognize patterns in sequential or time series data. These deep learning methods are frequently utilized for ordinal or temporal problems. This type of artificial neural network is characterized by its capability to handle sequences of different lengths. This makes them especially helpful in our scenario since they can utilize their internal memory to store information from previous inputs.

The concept of RNNs dates back to the 1980s. John Hopfield first introduced the Hopfield Network in 1982 [89]. However, RNNs began to gain significant attention in the 1990s when researchers started exploring their capabilities in depth, particularly for speech and handwriting recognition. Over the years, RNNs have evolved, with advancements addressing their initial limitations, such as difficulty in learning long-range dependencies within sequences.

The hidden state, also known as the memory state, is the main and most important feature of RNNs. It serves as the network's memory, allowing it to remember past information and affect future outputs. RNNs stand out from other neural network designs because of the hidden state's capacity to retain previous inputs. It keeps

track of the prior input sent to the network and uses the same parameters to carry out the same operation on all inputs or hidden layers in order to create the output. This greatly reduces the complexity of the parameters.

An RNN can be visualized in two ways: in its compact form and in its unrolled form. The compact form symbolizes the network's recurrent nature, while the unrolled form expands the network across the time dimension, illustrating how it processes a sequence step-by-step.

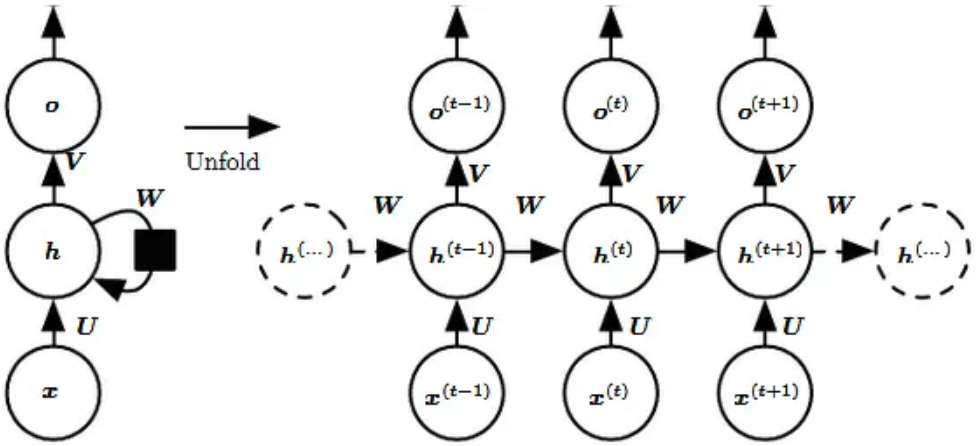


Figure 2.11: RNN Architecture [90].

- Input Layer: At each time step t, the network receives an input $x(t)$, such as a one-hot encoded vector representing a word in a sentence.

- Hidden Layer: The hidden state $h(t)$ at time t is a function of the current input and the previous hidden state $h(t-1)$, mathematically represented as $h(t) = f(Ux(t) + Wh(t-1))$. This computation incorporates a nonlinear activation function f, like tanh or ReLU, to introduce nonlinearity into the model.

- Weights: RNNs utilize three main weight matrices: U (input to hidden), W (hidden to hidden), and V (hidden to output). These weights are shared across

all time steps, significantly reducing the model's complexity by limiting the number of parameters to learn.

- Output Layer: The output at each time step $o(t)$ is derived from the hidden state, potentially passing through additional nonlinear transformations, especially if the network feeds into subsequent layers.

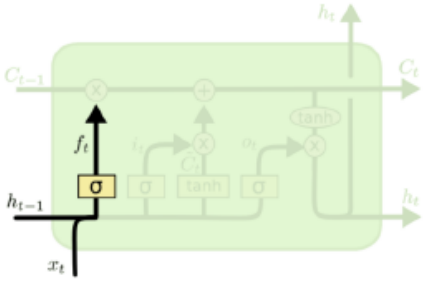
To address the limitations of basic RNNs, particularly their struggle with learning long-term dependencies due to the vanishing gradient problem, advanced variants like Long Short-Term Memory (LSTM) units and Gated Recurrent Units (GRUs) have been developed. These models introduce mechanisms to selectively remember and forget information, making them more effective for tasks requiring the understanding of long sequences.

LSTM

Long Short-Term Memory (LSTM) networks are a special kind of Recurrent Neural Network (RNN) that is well-suited for sequence prediction tasks due to their ability to learn long-term dependencies. LSTMs were introduced by Hochreiter and Schmidhuber in 1997 [92] to overcome the long-term dependency and the vanishing gradient problem that affects standard RNNs.

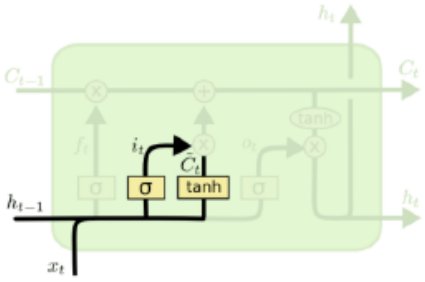
Recurrent Neural Networks (RNNs) are distinguished by their unique architecture. The main idea of this architecture is the concept of a chain of repeating modules, which allows RNNs to maintain a form of memory across the inputs they process. In standard RNNs, the repeating module within this chain-like structure is typically quite simple. This structure usually consists of a single layer with a nonlinear activation function, such as the hyperbolic tangent (tanh) function. However, this simplicity often brings some limitations. For example, even though RNNs are theoretically capable of handling long-range dependencies within sequences, in practice,

1.



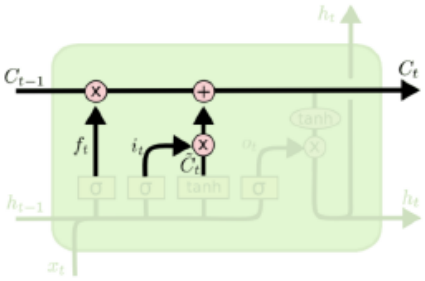
$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

2.



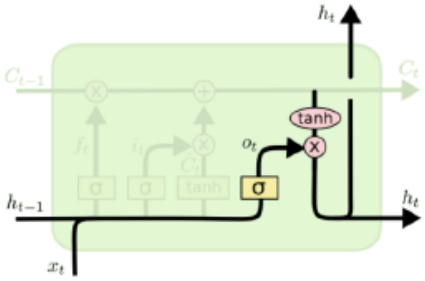
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

3.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

4.



$$o_t = \sigma\left(W_o \; [\, h_{t-1}, x_t] \; + \; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figure 2.12: LSTM Gates [91].

they struggle to maintain their internal state across long sequences due to issues like the vanishing gradient problem.

LSTMs make use of a more complex structure for the repeating module to learn these types of long-range dependencies. LSTMs use four distinct layers (Fig 2.12) instead of a single layer common in traditional RNNs. The fundamental principle of LSTMs is the cell state, which has few linear interactions and flows directly down the network's chain. This design allows information to flow unaltered and guarantees that data may be maintained and accessed by the network for extended periods of time.

1. **Forget Gate:** The forget gate decides which information should be discarded from the cell state. It looks at the previous hidden state h_{t-1} and the current input x_t and applies a sigmoid function to each number in the cell state C_{t-1}. The output of the function is a number between 0 and 1, where 0 means to "forget" and 1 means to "remember". This allows LSTM to retain only the important information necessary for future predictions.

2. **Input Gate:** The input gate is in charge of controlling the new information that goes to the cell state. This gate operates in a two-step process: first, a sigmoid layer called the "input gate layer" determines the values to be updated. Next, a tanh layer generates a vector of new candidate values $\tilde{C}_t$.

3. **Cell State:** The cell state functions as the LSTM's "memory." It carries the relevant information as the sequence is processed. It is modified by two gates. First, the input gate combines $\tilde{C}_t$ with the old state, effectively allowing the LSTM to update its memory with new relevant information. Second, the forget gate removes all useless information by multiplying the old state by f_t.

4. **Output Gate:** The output gate determines the next hidden state h_t based on the information from previous inputs. The hidden state can be used for

predictions and is carried over to the next time step. To determine which parts of the cell state to output, the output gate examines the previous hidden state and current input. This step is performed by employing a sigmoid function. Following this, a tanh function is applied to the cell state, resulting in a value between -1 and 1, which is then multiplied by the output of the sigmoid gate to produce only the chosen parts of the cell state.

The interaction among these four gates allows the LSTM to properly manage its internal state and output. This process makes it highly effective for tasks that require understanding sequences and their long-term dependencies. LSTM can make accurate predictions by choosing to forget irrelevant information, updating its memory with new information, and controlling the flow of information to the output. This complex process is what makes LSTMs the mainstay of sequence modeling in deep learning.

LSTM Sequence-to-Sequence

As discussed in the previous section, LSTMs work by mapping input sequences onto an output sequence. This mapping allows the model to capture the complex temporal patterns and dependencies within the data.

Predicting future values of a time series is a difficult task and one of the main topics of research in machine learning [93]. One of the most effective approaches for accomplishing this task involves the use of "sequence-to-sequence" (seq2seq) models. In this thesis, whenever we mention "sequence-to-sequence" we are specifically referring to many-to-many RNN seq2seq models. However, for the sake of brevity, we will simply use the term "sequence-to-sequence" throughout. This type of model is used in problems when the input and output sequences are of the same length. Seq2seq models are especially relevant if one of the objectives is to predict the future values of a dataset based on the information learned from its past values. Each element in the output sequence is directly predicted from a corresponding element in the input

sequence. Consequently, this method enables the model to maintain a continuous flow of information across the time steps.

More precisely, let $X = (x_1, x_2, \ldots, x_T)$ be the input sequence of length T and $Y = (y_{\tau+1}, y_{\tau+2}, \ldots, y_{\tau+T})$ be the corresponding output sequence. Here, each $y_{\tau+t}$ is the predicted value from x_t shifted by τ timesteps forward. At each timestep, t, the LSTM uses the input element x_t to update its hidden state, h_t, based on the previous hidden state h_{t-1}. Mathematically, this can be represented as:

$$h_t = f(h_{t-1}, x_t; \theta), \tag{2.43}$$

where f is the update function and θ represents the set of weights and biases that define the model.

Finally, once we have calculated the hidden state, we can proceed to calculate the output:

$$y_{\tau+t} = g(h_t; \phi), \tag{2.44}$$

where g is the output function and ϕ consists of the weights and biases that are part of the output layer of the model. These parameters control how the hidden states are translated into the output at each timestep.

2.3.4 Wavelet Transform

Wavelet transforms are a mathematical tool for analyzing data that provides a multiple time-scale representation of the system's dynamics. In this thesis, we calculate the amplitudes using the Morlet continuous wavelet transform for each postural mode [94]. Wavelets offer a multi-resolution time-frequency trade-off that allows for a more detailed depiction of postural dynamics across multiple time scales [95]. The wavelet

transform of a signal $y(t)$ at a time scale s is given by:

$$W_{s,\tau}[y(t)] = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} y(t)\psi^* \left(\frac{t-\tau}{s} \right) dt, \tag{2.45}$$

where $\psi(\eta)$ is the Morlet wavelet function, defined as:

$$\psi(\eta) = \pi^{-\frac{1}{4}} e^{i\omega_0 \eta} e^{-\frac{\eta^2}{2}}. \tag{2.46}$$

Here, $\omega_0 = 5$ is a non-dimensional parameter, and τ is a time point. Additionally, the time scale s is related to the Fourier frequency f by the equation:

$$s(f) = \frac{\omega_0 + \sqrt{2 + \omega_0^2}}{4\pi f} \tag{2.47}$$

This relationship is derived by maximizing the response of the wavelet transform to a pure sine wave, facilitating the interpretation of wavelet scales in terms of frequency.

The power spectrum $S(k, f; \tau)$ is calculated using:

$$S(k, f; \tau) = \frac{1}{C(s(f))} \left| W_{s(f),\tau}[y_k(t)] \right|, \tag{2.48}$$

where $C(s)$ is a scalar function for normalization:

$$C(s) = \frac{\pi^{-\frac{1}{4}}}{\sqrt{2s}} e^{1/4\left(\omega_0 - \sqrt{\omega_0^2+2}\right)^2}. \tag{2.49}$$

This normalization corrects for the Morlet wavelet's bias towards lower frequencies, ensuring a uniform response for all scales [27].

The frequencies for the wavelet spectrogram are chosen to be dyadically spaced between a minimum frequency $f_{\min} = 1$ Hz and the Nyquist $f_{\max} = 60$ Hz, using the formula:

$$f_i = f_{\max} 2^{-\frac{i-1}{N_f - 1} \log_2\left(\frac{f_{\max}}{f_{\min}}\right)}, \tag{2.50}$$

for $i = 1, 2, \ldots, N_f$, where N_f is the number of frequencies analyzed.

2.3.5 Autoencoders as a dimensionality reduction technique

High-dimensional spaces often suffer from the curse of dimensionality, where data points become increasingly isolated. This sparsity decreases the density of data, complicating the task of identifying accurate patterns or relationships among variables. The high dimensionality of the feature space means that conventional distance metrics become less informative. Therefore, in order to mitigate these challenges, traditional dimensionality reduction techniques like Principal Component Analysis (PCA) are usually employed. Linear methods like PCA reduce the dataset's dimensionality by projecting it onto a lower-dimensional subspace that captures the most variance. However, most of these methods fall short of portraying the full complexity of the data, especially when we encounter intricate nonlinear relationships.

Autoencoders, a class of neural networks designed for unsupervised learning (more in Section 2.2.7), can be used as a powerful alternative for dimensionality reduction. Because of their deep architecture, they can learn to encode and decode data in a way that both linear and nonlinear correlations are captured. This ability allows them to identify and represent the underlying structure of the data more accurately than linear methods. They compress the data into a lower-dimensional latent space, where the most critical features are preserved and the redundancies are eliminated.

The method begins with the training of the autoencoder. During this phase, the autoencoder adjusts its weights to minimize the difference between the original input data and its reconstruction from the latent representation. This involves learning the most efficient way to encode the salient features of the data into the latent space. Once the autoencoder has been adequately trained, the decoder component is removed. Now, we focus only on the encoding part. The output of the encoder, which is the projection of the high-dimensional data into the latent space, serves as a lower-

dimensional representation of the original data [96]. This latent representation is significantly easier to manage, visualize, and process.

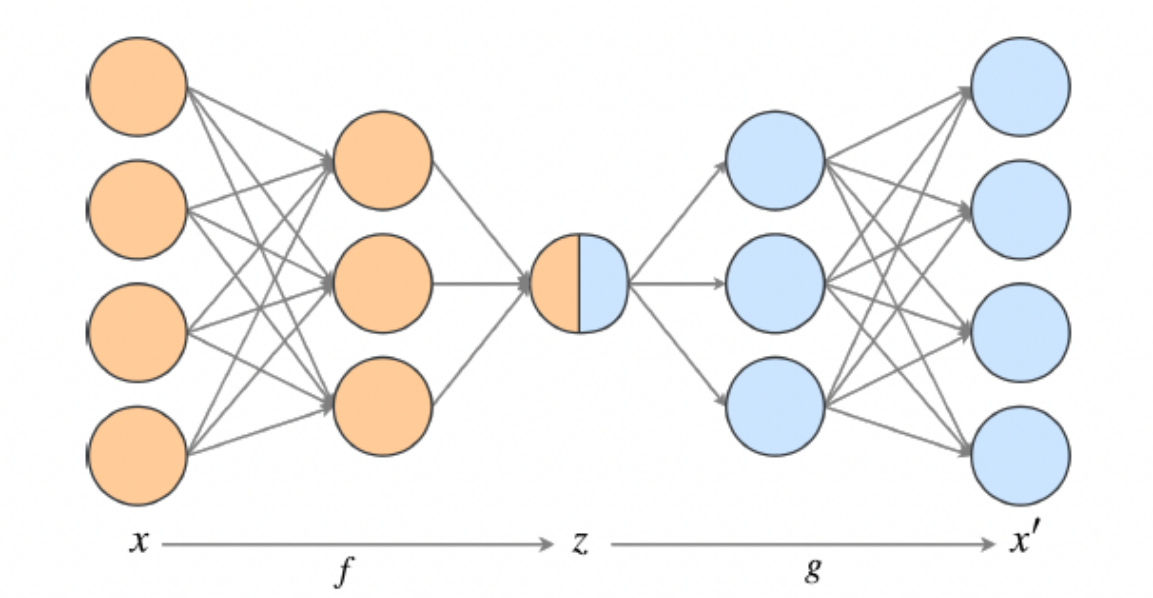


Figure 2.13: Deep autoencoder. The encoder is orange, and the decoder is blue. The reduced 1-dimensional representation is the bicolored node in the middle. Adapted from the comparative study by Fournier and Aloise on autoencoders versus traditional dimensionality reduction methods [96].

This approach effectively positions autoencoders as a non-linear generalization of PCA. While PCA linearly transforms data to a lower-dimensional space by projecting it onto the principal components that maximize variance, autoencoders go a step further. They use non-linear activation functions in their hidden layers to perform non-linear transformations to the original data in order to capture the data's inherent structure.

2.3.6 Spatial embedding

The approach taken to reduce the dimensions of the fixed points identified by the RNN must be designed to reduce distortions in the local embedding, ensuring that it accurately captures close relationships in the data. This strategy should not priori-

tize the preservation of global structure or long-distance relationships, providing more flexibility in representing distances between distant points on the manifold. Therefore, we selected t-Distributed Stochastic Neighbor Embedding (t-SNE) since it is a method that possesses all these properties [97]. Other traditional methods, such as PCA, multi-dimensional scaling, and Isomap do exactly the opposite, sacrificing local accuracy to obtain better global structure [98, 99, 100].

t-SNE is a popular method for dimensionality reduction that is particularly well-suited for visualizing high-dimensional datasets. It works by converting distances between data points in the high-dimensional space into conditional probabilities that represent similarities. The similarity of data point t_j to data point $t_j i$ is given by a conditional probability $p_{j|i}$, which is defined as:

$$p_{j|i} = \frac{\exp\left(-\frac{d(t_i,t_j)^2}{2\sigma_i^2}\right)}{\sum_{k\neq i}\exp\left(-\frac{d(t_i,t_k)^2}{2\sigma_i^2}\right)}, \tag{2.51}$$

where $d(t_i, t_j)$ is the distance between two points t_i and t_j, and σ_i is the variance of the Gaussian that is centered on data point t_i. The conditional probability $p_{j|i} = 0$, since all self-transitions are excluded. Similarly, in the low dimensional space, a new set of transition probabilities $q_{j|i}$ are defined. However, in this case, it is important that these pairwise similarities are proportional to a Student-t kernel of the points' Euclidean distances in the embedded space.

Finally, a distance function must be defined. We aim to minimize the differences between $p_{j|i}$ and $q_{j|i}$; thus, a reasonable distance function is the Kullback-Leibler divergence (KLD) between the two distributions [101].

Chapter 3

Inferring time-varying coupling of dynamical systems with temporal convolutional network autoencoders

3.1 Introduction

Rather than being static in time, interactions between parts of a complex system continuously ebb and flow, with one variable driving another at one point, just to see the relationship reverse or lessen or disappear at a later point in time. Real-world signals are seldom stationary and well-behaved, and their causal linkages and interactions frequently appear, disappear, and reappear, possibly changing in strength over time. Examples of such systems abound in neuroscience [3, 102], ecology [38], finance, [8, 37, 6], and climate [103, 104, 40].

Despite the ubiquity of these dynamically altering interactions, however, most methods for assessing causality in complex dynamical systems have difficulty mea-

suring how the direction and extent of interactions between variables in a system alter in time. This difficulty arises from several inherent characteristics of complex systems and the limitations of existing causal assessment methodologies. Most of these methods, including Granger Causality (GC) [51], often assume that the dynamical system should be approximately stationary, meaning their statistical properties do not change over time. Other common assumptions, such as linearity and time-invariance are also often violated in complex real-world systems. These constraints significantly limit the applicability and accuracy of these approaches in many scenarios [16, 81, 105, 7, 106, 4].

In addition, in systems where variables are strongly coupled and synchronized, some of these causality inference methods struggle to accurately infer the coupling strength and direction of causality [4]. This issue extends to scenarios of intermediate coupling, where the variables are neither weakly nor strongly linked. Additionally, the presence of noise in the system leads to a decrease in cross-mapping fidelity, revealing further limitations [107, 108]. Although the lack of correlation is neither necessary nor sufficient to demonstrate causation [35, 36], correlation does play an important role in many statistical methods as the basis for hypothesis tests for causality. Mirage correlations can appear in the simplest nonlinear systems [34]. Variables that may be positively correlated at some point in time can become anti-correlated some moments after or even lose all coherence. However, most causality methods do not adequately account for the fact that sudden changes in correlation over time between variables may indicate a change in the underlying temporal causal relationships.

In this chapter, we introduce a new methodology for probing time-varying causal interactions using a new metric for assessing causal interactions combined with a novel machine learning architecture for causal inference, which we call Temporal Autoencoders for Causal Inference (TACI). We show the method's effectiveness on synthetic and real-world data sets, both in an absolute sense and in comparison to

extant methods, particularly focused on how to find time-varying causal structure in complex dynamical systems.

3.2 Overview of Methodology

In our methodology, we adopt a two-fold approach towards developing a causal inference method that accurately assesses causality between variables $x(t)$ and $y(t)$ in the Granger sense for nonlinear systems with time-varying interactions. The first aspect of our approach is to use a novel surrogate data comparison metric – the Comparative Surrogate Granger Index (CSGI) – that measures the relative improvement in prediction accuracy when including both variables vs. one of them and a randomized version of the other. The other aspect is to use a two-headed Temporal Convolutional Network architecture to robustly capture the space of potential nonlinear mappings between variables across the entire time series. As will be observed, the CSGI with linear autoregressive models works well to identify causal interactions in situations where relatively straightforward mappings exist between variables, but the more complicated neural network model is more effective when the mappings are more non-linear.

3.2.1 Comparative Surrogate Granger Index (CSGI)

Informally, Granger Causality (GC) defines a causal interaction from $x(t)$ to $y(t)$ to be when knowing the full history of both $x(t)$ and $y(t)$ provides a better prediction about the future of $y(t)$ than knowing just the history of $y(t)$ alone. While there are many variations on this methodology [24, 109, 76], the typical form used is to compare two models of similar type (e.g., linear autoregressive models, feedforward neural networks, etc.) based on their ability to predict the future state of $y(t)$. More

explicitly, the comparison is between

$$y(t) \approx f(y_{t-1}, y_{t-2}, \ldots) \tag{3.1}$$

and

$$y(t) \approx g(x_{t-1}, x_{t-2}, \ldots, y_{t-1}, y_{t-2}, \ldots). \tag{3.2}$$

Usually, an F-test is used to determine whether the latter model is preferred over the former.

This form, however, suffers from two limitations. First, the comparison is a binary one – the second model is "significantly" better than the first or it is not – thus, differences in the strength of coupling can not be detected, just the presence or absence. Second, because the F-test and similar methods incorporate strong assumptions about the underlying dynamics of the system, statistical statements deriving from these tests are often not robust under resampling or re-parameterization. In addition, because the model complexities for the two models being compared are inevitably quite different, with one typically having twice as many parameters as the other, the F-test often fails to detect causal interactions properly.

A common strategy for ameliorating these limitations is to compare not

$$f(y_{t-1}, y_{t-2}, \ldots) \text{ and } g(x_{t-1}, x_{t-2}, \ldots, y_{t-1}, y_{t-2}, \ldots), \tag{3.3}$$

but rather (as described in Eqns. (3.1) and (3.2))

$$f(x_{t-1}, x_{t-2}, \ldots, y_{t-1}, y_{t-2}, \ldots) \text{ and } g(x^{(s)}_{t-1}, x^{(s)}_{t-2}, \ldots, y_{t-1}, y_{t-2}, \ldots), \tag{3.4}$$

where $x^{(s)}$ is a *surrogate* data set that shares similar statistical properties to $x(t)$ but is shuffled in some manner (e.g., shuffling values to preserve the distribution of values or shuffling phases of the time series' Fourier Transform to preserve the frequency

profile). Typically, this comparison is accomplished through the Extended Granger Causality Index (EGCI) [24, 52]. If $\epsilon_y(t)$ are the residuals for fitting the future of $y(t)$ on the past of $y(t)$ and $\epsilon_{xy}(t)$ are the residuals for fitting the future of $y(t)$ on the pasts of both $x(t)$ and $y(t)$, then the EGCI is given by ratio between the relative reduction in residual variance when the past of $x(t)$ is included in the model:

$$\text{EGCI} = 1 - \frac{\text{var}(\epsilon_{xy}(t))}{\text{var}(\epsilon_y(t))}. \tag{3.5}$$

$y(t)$ is thus said to cause $x(t)$ if the EGCI using the actual values of $x(t)$ is significantly higher than the EGCI found substituting $x(t)$ for $x^{(s)}(t)$.

Our approach attempts to assess directly the relative increase in variance explained for the predictive model when using $x(t)$ vs. $x^{(s)}(t)$. Specifically, if R^2_{xy} is the fraction of variance explained about the future of $y(t)$ using the pasts of $x(t)$ and $y(t)$ in the model and $R^2_{x^{(s)}y}$ is the fraction of variance explained using $x^{(s)}(t)$ and $y(t)$, then we define the Comparative Surrogate Granger Index (CGSI), $\chi_{x\to y}$, to be defined via

$$\chi_{x\to y} = \frac{R^2_{xy} - R^2_{x^{(s)}y}}{\frac{1}{2}(R^2_{xy} + R^2_{x^{(s)}y})}. \tag{3.6}$$

This metric's advantages over the EGCI are that it is able to measure small changes in causal interactions and that it explicitly measures the difference in predictive power between using actual data and using surrogate data to predict the future. In this chapter, we will be measuring $\chi_{x\to y}$ and $\chi_{y\to x}$ for all pairs of variables to assess whether there is causal coupling between two variables, whether it is uni- or bi-directional, and the relative strength of the coupling.

3.2.2 Temporal Autoencoders for Causal Inference

While one advance in our methodology is the use of the CGSI in the previous section, the other novel contribution is the use of a new artificial neural network architecture

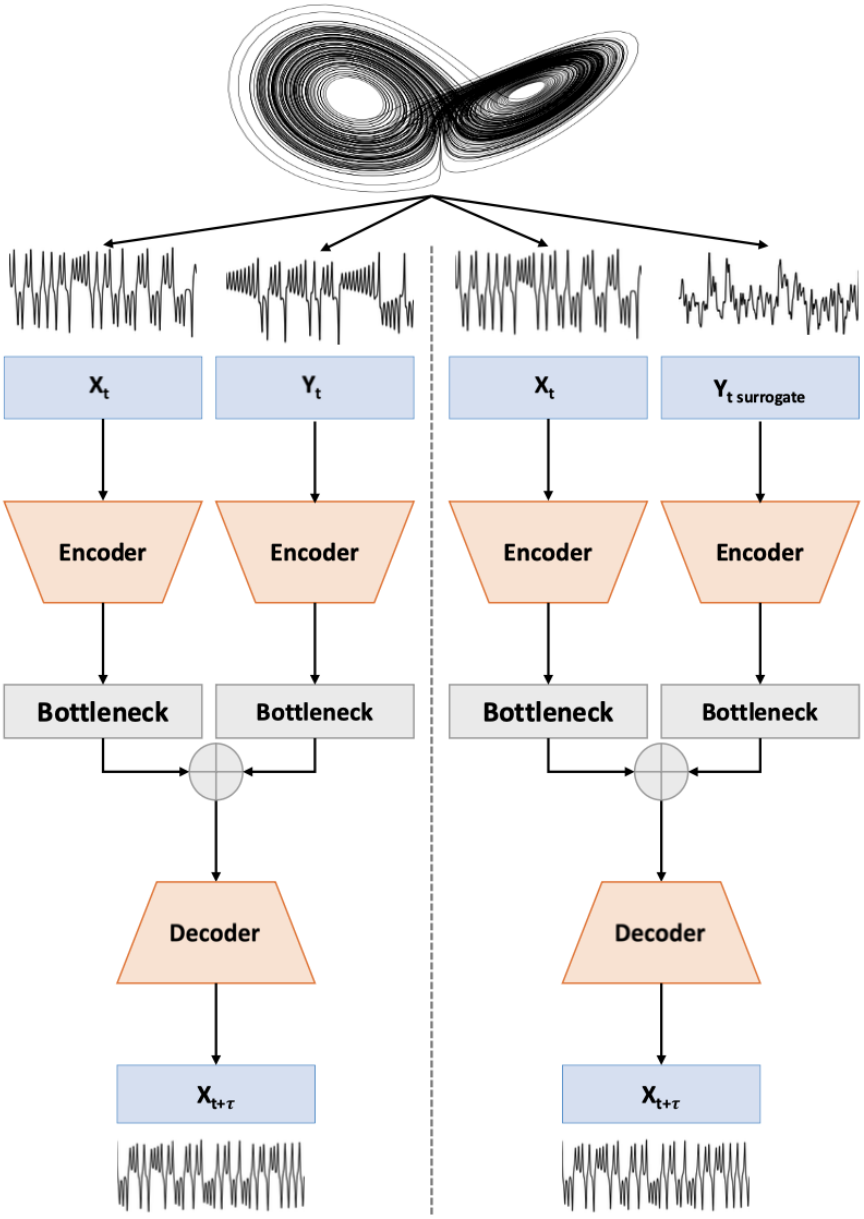


Figure 3.1: Schematic of the Temporal Autoencoders for Causal Inference (TACI) Networks. We use a two-headed network consisting of Temporal Convolutional Networks that interact through a shared latent space to predict a time-shifted version of one of the two input time series. For each pair of variables we wish to examine (here, X and Y), we train two networks for each causal direction: one using X and Y as inputs and another using X and a Fourier-shuffled surrogate version of Y. We consider an interaction from $Y \to X$ to be causal if the network using the actual value of Y predicts the future of X better than the network using the surrogate version of Y. In this particular case, we show the approach applied to two different variables from the Lorenz system.

to calculate the functions f and g in Eqn. (3.4) that are used to predict the future of $y(t)$. The original (and still most common) models ([51]) for f and g are auto-regressive linear models of the form

$$y(t) = \sum_{i=1}^{k} (a_i x(t-i) + b_i y(t-i)) + \epsilon_t, \tag{3.7}$$

where $x^{(s)}(t)$ can be substituted for $x(t)$ when using the surrogate approach. While this relatively simple approach shows impressive performance in a variety of scenarios, these models fail to accurately predict known causal interactions for couplings that have weak to moderate coupling and are governed by nonlinear dynamics that are not well approximated by linear models [21, 17]. This inability is typically because the systems fail to satisfy separability. In other words, all information about a causative factor has to be inherent to that specific variable and can be omitted by removing that variable from the model, as is the case for purely stochastic or linear systems [4]. For systems with strongly nonlinear deterministic components, however, this assumption fails, and, accordingly, so do the predictions from auto-regressive linear GC [105, 17]. In addition, because these linear models have difficulty predicting information across multiple timescales, they often have difficulty detecting subtle shifts in causality as a function of time.

In recent years, a solution has been to replace the linear model in (3.7) with deep neural networks of varying architectures that, due to their nearly non-parametric nature, excel in approximating complex functions [90]. These methods include the use of Variational Autoencoders to estimate causal effects [110], Causal Generative Neural Networks to learn functional causal models [111], Neural Granger to estimate non-linearly dependencies based on Granger causality principles [76], and the Temporal Causal Discovery Framework (TCDF) to address time delay causal relationships [81]. These methods, however, are often unwieldy to train, are prone to overfitting, and

are susceptible to inaccuracies in the presence of a significant amount of noise.

In this chapter, we introduce a novel neural network architecture for causality using a two-headed Temporal Convolutional Network (TCN) autoencoder (Fig. 3.1). TCNs, the primary building block of our approach, are a specialized type of neural network that integrates causal inference into convolutional architectures. First introduced for video-based action segmentation [78], TCNs quickly became popular due to their ability to extend most of the convolutional advantages of regular CNNs – including sparsity and translational equivariance – into the time domain through a series of dilated and causal convolutional layers. A key characteristic of this framework is its simplicity, relatively long memory, and ability to outperform most convolutional architectures in auto-regressive prediction tasks [79].

Our approach, which we call Temporal Autoencoders for Causal Inference (TACI) is a neural network that consists of a two-headed TCN autoencoder, where two TCNs are used to encode time series $x(t)$ and $y(t)$, and a third is used for decoding an equivalently long time series describing the future trajectory of $y(t)$ (shifted by some time, τ) from a relatively low-dimensional latent space that is derived from the outputs of the first two autoencoders. A more detailed description of our model and our training methodology can be found in *Materials and methods*. Code is available here: https://github.com/josuancalderonglez/Temporal-Autoencoders-For-Causal-Inference-TACI-/tree/main.

For each comparison of interest, we train four versions of this network: one using $x(t)$ and $y(t)$ as input time series to predict the future of $x(t)$, another that is the same except for replacing $x(t)$ with the surrogate data $x^{(s)}(t)$, another pair of the networks that are structured the same except with x and y reversed in each case. Given these four trained networks, we can then make predictions for the future of the appropriate variable and calculate the fraction of variance explained over a moving window (i.e., $R^2_{xy}(t)$ or $R^2_{x^{(s)}y}(t)$). From these values, we can then apply Eqn. (3.6) to

calculate the CGSI values $\chi_{x\to y}$ and $\chi_{y\to x}$, which we will use to assess causal inference between these two variables.

3.2.3 Other Methods We Compare Against

Alongside comparing GC with linear autoregressive models, which we will refer to here as Surrogate Linear Granger Causality (SLGC), we will also test against two other commonly used methods: Convergent Cross Mapping and Transfer Entropy.

Convergent Cross Mapping (CCM)

Convergent Cross Mapping (CCM) was introduced to determine causation in systems that could be modeled as relatively noiseless deterministic dynamical systems [4]. The core concept of this approach is that according to Takens' Embedding Theorem, if $x(t)$ and $y(t)$ are two variables of a deterministic dynamical system, one can reconstruct $x(t)$ from a delay embedding of $y(t)$ if and only if the time derivative of $x(t)$ explicitly depends on $y(t)$ [64, 65, 19]. Thus, if it is possible to predict $x(t)$ from an embedding of $y(t)$ alone, we would say that y has a causal interaction with x. Practically, these predictions are calculated by predicting $x(t)$ from $y(t)$ (and vice versa) and computing the correlation coefficient between the actual and predicted values [4], with correlations near one implying a strong casual influence and correlations near zero implying no or little influence. Here, we used Scikit Convergent Cross Mapping (skccm), a Python-based-library implementation of CCM for causal discovery [112].

Transfer Entropy

Transfer entropy (TE) is a metric that quantifies a reduction in uncertainty in predicting the future of one variable given the past of another using formalism from information theory [22]. Specifically, we can measure the transfer entropy from $y(t)$

to $x(t)$ at a given distance in the future, τ, ($T_{Y\to X}(\tau)$) via

$$T_{Y\to X}(\tau) = H(x(t)|x(t-1),\ldots,x(t-\tau)) \\ - H(x(t)|x(t-1),\ldots,x(t-\tau),y(t-1),\ldots,y(t-\tau)), \tag{3.8}$$

where $H(X|Y)$ is the Shannon entropy of the conditional probability distribution $p(X|Y)$. This quantity is zero if adding information about the past of $y(t)$ results in no reduction in our future guesses for $x(t)$, and if it is non-zero, the quantity can be interpreted as the rate of information flowing from Y to X. Practically, we calculate TE for our systems using the Java Information Dynamics Toolkit (JIDT or Infodynamics Toolkit) [113].

3.3 Results

3.3.1 Artificial Test Systems

To test the validity of our approach, we applied the methodology to a variety of different deterministic and stochastic dynamical system models with known causal interactions, finding that TACI performs well across all cases. In particular, we are interested in cases where the coupling changes in time, which we will explore in detail for the Coupled Henon Maps system.

The Rössler-Lorenz System

Our first example case is a system of coupled chaotic attractors, where the Lorenz system ($\vec{y}$) [114] is driven by a Rössler oscillator ($\vec{x}$)[115]:

$$\begin{aligned}
\dot{x}_1 &= -6(x_2 + x_3), \\
\dot{x}_2 &= 6(x_1 + 0.2x_2), \\
\dot{x}_3 &= 6\Big(0.2 + x_3(x_1 - 5.7)\Big), \\
\dot{y}_1 &= 10(y_2 - y_1), \\
\dot{y}_2 &= 28y_1 - y_2 - y_1y_3 + Cx_2^2, \\
\dot{y}_3 &= y_1y_2 - \frac{8}{3}y_3,
\end{aligned} \tag{3.9}$$

where the constant C controls the coupling strength of the system, and the driving severely distorts the behavior of the Lorenz attractor as C increases [116]. Synchronization between $\vec{x}$ and $\vec{y}$ starts near $C = 2.14$, making the two systems' behavior effectively coupled above this point despite the lack of an explicit coupling term, making traditional formal notions of causality ill-posed (Fig. 3.2A) [39]. The solutions to the differential equations were generated by using a fourth-order Runge-Kutta method. C was chosen between 0 and 5, computing a time series of length 300,000 ($dt = 0.1$) after a burn-in time of 30,000 time points at each coupling strength.

As seen in Figure 3.2B-E, TACI is the only method of the four tried here that accurately predicts the unidirectional coupling from $\vec{x}$ to $\vec{y}$. SLGC (Fig. 3.2B) fails to predict any coupling whatsoever between the variables, and CCM and TE (Figs. 3.2C-D) predict bidirectional coupling (albeit with somewhat more information flowing from $\vec{x} \rightarrow \vec{y}$ than in the reverse). TACI, in contrast, predicts only unidirectional coupling until the point of synchronization ($C \approx 2.14$), after which, it predicts no effective causation in either direction.

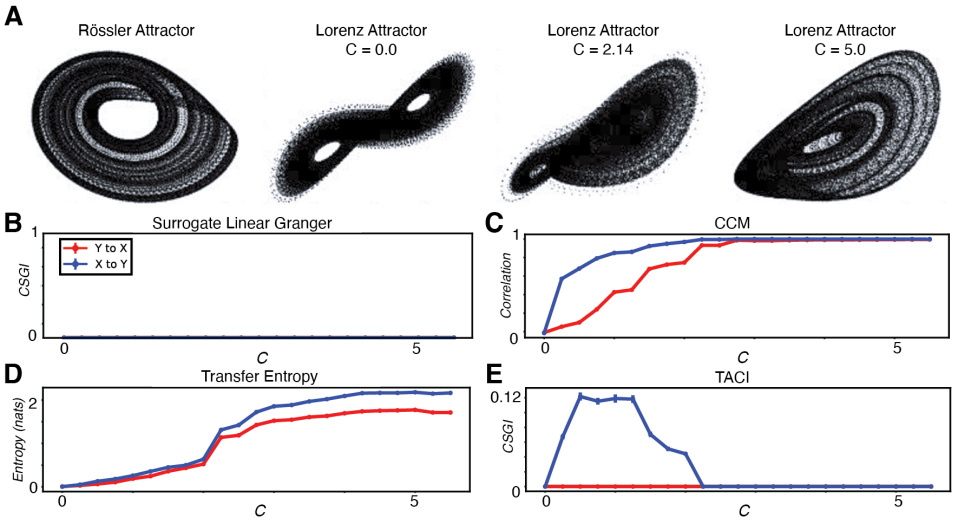


Figure 3.2: Causal inference in the Rössler-Lorenz System. **A**) 2-dimensional projections of the Rössler attractor (left) and the Lorenz system (right three plots) as C increases. Mathematically, there is only coupling from $X \to Y$, but starting near $C = 2.14$, the two systems become synchronized, making finding the causal interactions an ill-posed problem. **B-E**) Results from applying the four methods to the system. Note that only TACI accurately predicts the unidirectional coupling in the regime above $C > 0$ and before synchronization occurs. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

Coupled Bi-directional Two-Species Model

In contrast to the Rössler-Lorenz System, the bi-directional two-species model [117], is calculated in discrete time, and it exhibits (unsurprisingly) bi-directional coupling:

$$\begin{aligned} x(t+1) &= x(t)\Big(3.8 - 3.8x(t) - Cy(t)\Big), \\ y(t+1) &= y(t)\Big(3.5 - 3.5y(t) - 5Cx(t)\Big), \end{aligned} \tag{3.10}$$

where C once again is the coupling strength, noting that the coupling strength is five times larger from $x \to y$ than in the reverse direction. In this system, separability is not satisfied (i.e., information about y is redundantly present in x and vice versa). Despite the fact this model is deterministic and dynamically coupled, it shows alternating periods of positive, negative, and zero correlation [4]. For values of $C \in [0, 0.35]$, we created a bivariate time series of length 300,000 (after a burn-in

of 30,000 time points). The initial conditions were generated with random starting points drawn from the uniform distribution $(0.01, 0.99)$.

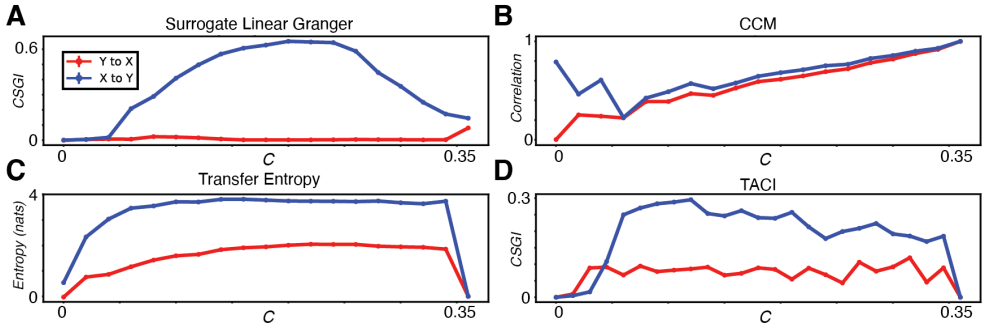


Figure 3.3: Causal inference in the bidirectional species system. **A-D**) Results from applying the four methods to the bidirectional species system. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

Applying the four methods to these data (Fig. 3.3), we find that both TE and TACI correctly identify both the bi-directional aspect of the coupling and the increased causal link from $x \to y$ compared to $y \to x$. CCM identifies the bi-directionality correctly, but it does not identify the relative strength of the couplings, and SLGC is unable to identify any causal link from $y \to x$.

Coupled Autoregressive Models

Coupled autoregressive models are an extension of basic autoregressive models, intended to represent the dynamics of systems where multiple time series influence each other. In these models, the value of a variable at a given time point is not only a function of its own previous values but also depends on the past values of other variables in the system. Here, we study the following system consisting of two bidirectionally coupled autoregressive processes of the first order:

$$\begin{aligned} x(t+1) &= 0.5x(t) + 0.2y(t) + \epsilon_x(t), \\ y(t+1) &= Cx(t) + 0.7y(t) + \epsilon_y(t), \end{aligned} \tag{3.11}$$

where C is the strength of the coupling between x and y, and $\epsilon_x(t)$ and $\epsilon_y(t)$ are drawn from a normal (Gaussian) distribution with a mean of 0 and $\sigma_x^2 = \sigma_y^2 = 0.1$. Higher values of C represent stronger couplings from $x \to y$, and for $C = 0$, the system is unidirectional (only the past of y has an impact on the future of x). We examined values of $C \in [0, 0.6]$ and created sets of bivariate time series of length $L = 300,000$ for each value of C (after a burn-in time of $30,000$ points). The initial conditions of the system were generated from the normal distribution with zero mean and unit variance.

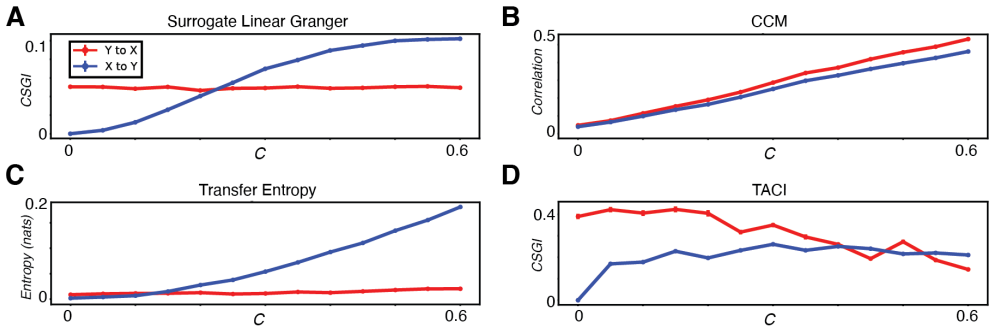


Figure 3.4: Causal inference in the coupled autoregressive models system. **A-D**) Results from applying the four methods to the coupled autoregressive models system. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

Fig. 3.4 shows that SLGC does very well at identifying the onset of bi-directionality for $C > 0$, with the coupling of $x \to y$ monotonically increasing with C. This fact is perhaps not surprising, as SLGC is based on precisely such linear systems. TACI also does a comparable job at detecting bi-directionality, even roughly predicting the switchover between $x \to y$ and $y \to x$ coupling strengths at $C = 0.5$. CCM, however, does not predict any coupling from $y \to x$ at $C = 0$, and TE does not predict any coupling from $y \to x$ across all values of C.

Coupled Hénon Maps

Our last example, the Hénon map, is a well-known example of a discrete-time dynamical system that exhibits chaotic behavior that was first developed as a simplified version of the Poincaré portion of the Lorenz model [117], and in its chaotic regime, it is characterized by an attractor with a warped horseshoe shape. Here we consider a case of two Hénon maps, $\vec{x}$ and $\vec{y}$, with unidirectional coupling [118]:

$$\begin{aligned} x_1(t+1) &= 1.4 - x_1^2(t) + 0.3x_2(t), \\ x_2(t+1) &= x_1(t), \\ y_1(t+1) &= 1.4 - \big(Cx_1(t)y_1(t) + (1-C)y_1^2(t)\big) + 0.3y_2(t), \\ y_2(t+1) &= y_1(t), \end{aligned} \tag{3.12}$$

where C controls the strength of the coupling from $\vec{x}$ to $\vec{y}$. For coupling strengths above $C > 0.65$, the systems start to show evidence of intermittent synchronizations. This on-off behavior becomes a fully synchronized state after $C > 0.7$ [119]. For $C \in [0, 0.9]$, we generated sequences of length 300,000 (after a burn-in period of 30,000) and analyzed data from x_1 and y_1 for each of the methods. TACI is the only

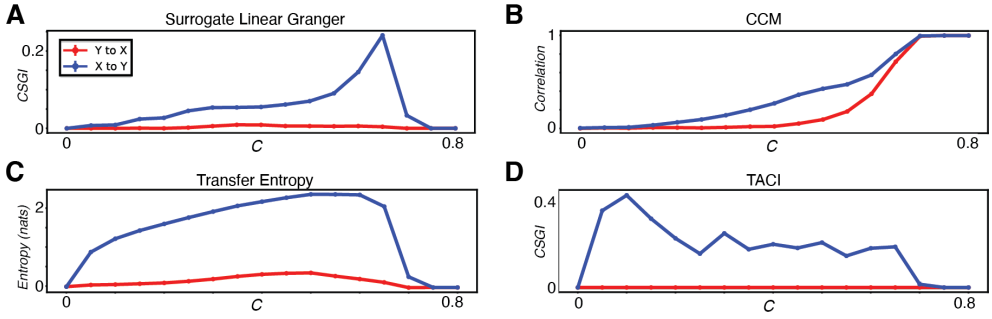


Figure 3.5: Causal inference in the coupled Hénon Maps system. **A-D**) Results from applying the four methods to the coupled Hénon Maps system. Here, only TACI accurately predicts univariate coupling across all values of C prior to synchronization. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

method out of the four that correctly identifies the uni-directional coupling between from $\vec{x}$ to $\vec{y}$ (but not from $\vec{y}$ to $\vec{x}$), although SLGC is very close, it is statistically significantly different from zero at intermediate values of C. TE and CCM both predict bi-directional interactions, albeit with weaker coupling from $\vec{y}$ to $\vec{x}$ than in the reverse direction.

Non-stationary Coupled Hénon Maps

Taken as a whole, TACI is the only method that performed well across all four artificial test cases, but the challenge remains as to whether it can identify patterns in data that change over time. To test this idea, we generated time series from the coupled Hénon maps in (3.12) but with time-varying couplings, $C_{xy}(t)$ and $C_{yx}(t)$:

$$\begin{aligned} x_1(t+1) &= 1.4 - \left(C_{yx}(t) y_1(t)\, x_1(t) + (1 - C_{yx}(t))\, x_1^2(t)\right) + 0.3 x_2(t)\,, \\ x_2(t+1) &= x_1(t)\,, \\ y_1(t+1) &= 1.4 - \left(C_{xy}(t) x_1(t)\, y_1(t) + (1 - C_{xy}(t))\, y_1^2(t)\right) + 0.3 y_2(t)\,, \\ y_2(t+1) &= y_1(t)\,. \end{aligned} \tag{3.13}$$

Here, the two coupling terms are similar to the coupling term, C, in (3.12) but with time-varying values and potentially allowing for coupling from y to x.

We performed four different tests to see how TACI performs when causal interactions alter with time: *(i)* setting $C_{yx}(t) = 0$ and toggling $C_{xy}(t)$ between 0 and 0.6, *(ii)* initially setting $C_{yx}(t) = 0$ and $C_{xy}(t) = 0.6$ and then switching the two half-way through the run, *(iii)* setting $C_{yx}(t) = 0$ and toggling $C_{xy}(t)$ between 0 and 0.6 but with pulses of $C_{xy}(t) = 0.6$ being set to different time widths, and *(iv)* setting $C_{yx}(t) = 0$ and stepping $C_{xy}(t)$ from 0 to 0.4 and back down again in steps of 0.1.

Other than the coupling changes, all time series were generated in an identical manner to the previous section. It is important to note that the network for TACI was

only trained once on the entire time series, not specifically for each testing window. Thus, by creating a robust model, our network is able to identify complex causal dynamics that change in time without having to constantly fit new models, as would be the case for SLGC, CCM, and TE.

In Fig. 3.6, we show that TACI performs well in the first three of these scenarios, ably identifying when eliminations of causal interactions occur, as well as when $C_{yx}(t) = 0$ and $C_{xy}(t)$ flip. In addition, Fig. 3.7 shows that the TACI network is able to identify how coupling strengths change with time.

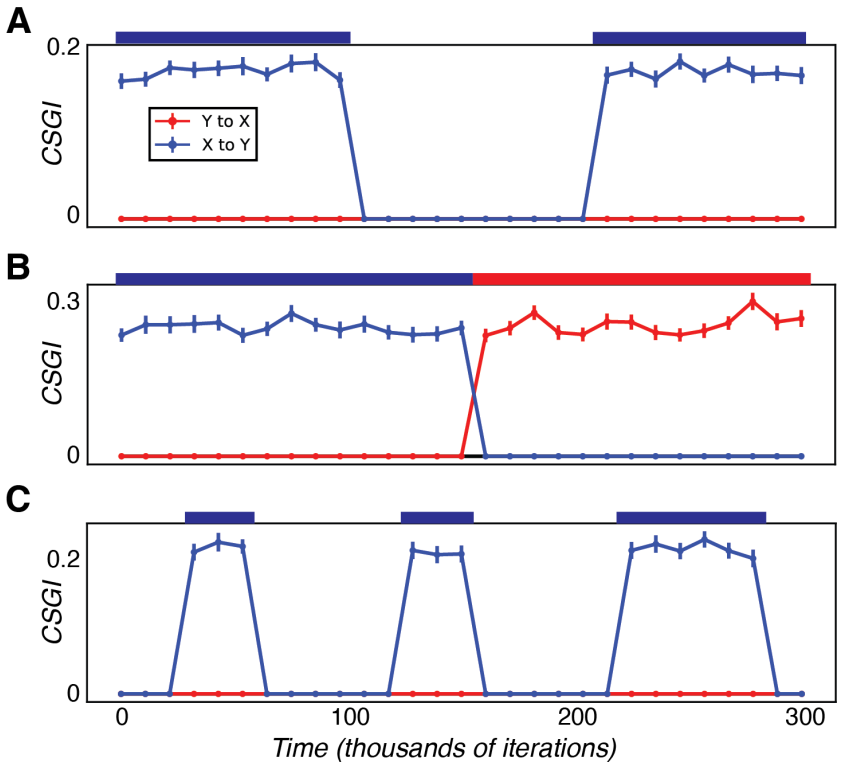


Figure 3.6: TACI applied to coupled non-stationary Hénon Maps. **A**) A plot of the TACI inference when applied to the coupled Hénon Maps system where the coupling from $X \to Y$ is set to either $C_{xy} = 0.6$ (blue bar above the plot) or $C_{xy} = 0$ (no bar above the plot). **B**) Same as **A** but with a toggle from $C_{xy} = 0.6$ to $C_{yx} = 0.6$ (where the blue and red bars above the plot flip). **C**) Same as **A** but with multiple pulses of $C_{xy} = 0.6$ of varying sizes. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

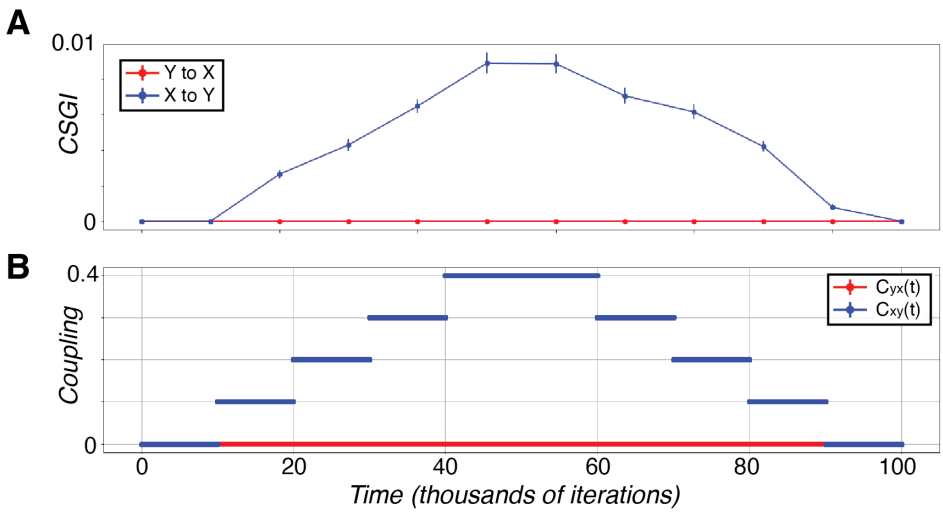


Figure 3.7: TACI applied to coupled non-stationary Hénon Maps with ramped couplings. **A**) Inferred causal coupling as a function of time during the simulation. **B** Time series of how the coupling from X to Y was stepped up and then down. Error bars are generated using a bootstrapping procedure (see Materials and Methods).

Summary of Results on Artificial Test Systems Among the methods tested, only TACI is able to robustly infer known causal interactions between variables without incorrectly predicting non-existent interactions. TACI consistently differentiates between unidirectional and bidirectional coupling in low, moderate, and strong settings. Additionally, it accurately detects instances when the time series become synchronized in all tested scenarios. TACI excels in identifying complex causal dynamics that evolve over time, such as those observed in pulse systems with time-varying coupling. Given these successes in artificial systems, we will now apply the method to two real-world examples.

3.3.2 Jena Climate Dataset

The first data set we will test our model on is the "Jena Climate Dataset", a detailed collection of weather measurements recorded by the Max Planck Institute for Biogeochemistry from a weather station located in Jena, Germany [120]. The dataset spans nearly eight years – from January 10, 2009, to December 31, 2016 – and includes 14

distinct meteorological features recorded every 10 minutes. These features include a wide range of atmospheric conditions, from temperature to relative humidity to vapor pressure deficit (see Table 3.1 for details). Several example time series are shown in Fig. 3.8.

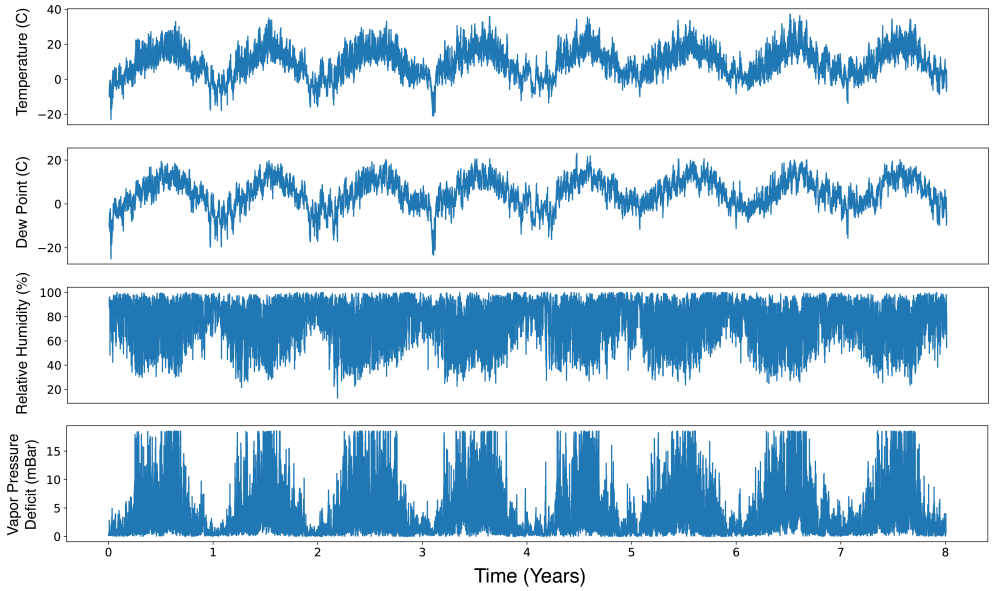


Figure 3.8: Time series of Temperature, Dew Point, Relative Humidity, and Vapor Pressure Deficit from the Jena Climate Dataset.

A key advantage of these data is that some of the interactions are known already due to empirical models of atmospheric dynamics, providing a good test case for our method on real data. One example is the relationship between relative humidity (R_H), the dew point (T_{dew}), and the temperature (T), which is given by

$$R_H = 100 \exp \frac{17.625 T_{dew}}{T_{dew} + 243.04} \Big/ \exp \frac{17.625 T}{T + 243.04}, \tag{3.14}$$

where T_{dew} and T are in degrees Celsius and R_H is a percentage [121]. Calculating the partial derivative of R_H with respect to T (keeping T_{dew} fixed), we find that we should expect stronger interactions to occur from T to R_H at lower temperatures (Fig. 3.9A). After training our TACI model from each of the variables in the data set onto T, we indeed find that causal interactions peak during epochs when the temperature

Table 3.1: Summary of Jena Climate Dataset Features

Feature	Description
Date/Time	Date-time reference
p (mbar)	Atmospheric pressure stated millibars
T (degC)	Temperature in Celsius
Tpot (K)	Temperature in Kelvin
T_{dew} (C)	Dew Point Temperature in Celsius
R_H (%)	Relative Humidity in percentage
VPmax (mbar)	Saturation vapor pressure
VPact (mbar)	Vapor pressure
VPdef (mbar)	Vapor pressure deficit
sh (g/kg)	Specific humidity
H_2O C (mmol/mol)	Water vapor concentration
rho (g/m^3)	Air density
wv (m/s)	Wind speed
max. wv (m/s)	Maximum wind speed
wd (deg)	Wind direction in degrees

drops (Fig. 3.9B), showing that our method can accurately find temporal variations in causal interactions in messy real-world data.

3.3.3 Electrocorticography in Non-Human Primates

Lastly, we used electrocorticography (ECoG) data from non-human primates to test whether our methodology can detect time-varying interactions between brain regions from these electrophysiological signals. These data frequently exhibit extraordinarily complex dynamics that shift in time as an animal changes its state: from sleep to wake, from satiated to hungry, from attending from one object to another, and so on [102]. These alterations are often subtle, and, thus, understanding how different regions of the brain drive one another's activity requires a method that can detect how slight variations in the relationship between variables lead to changing interactions across time.

Here, we analyzed publicly available ECoG data from a single monkey (*Macaca fuscata*) [122, 123, 124]. These recordings consisted of 128 channels of data that

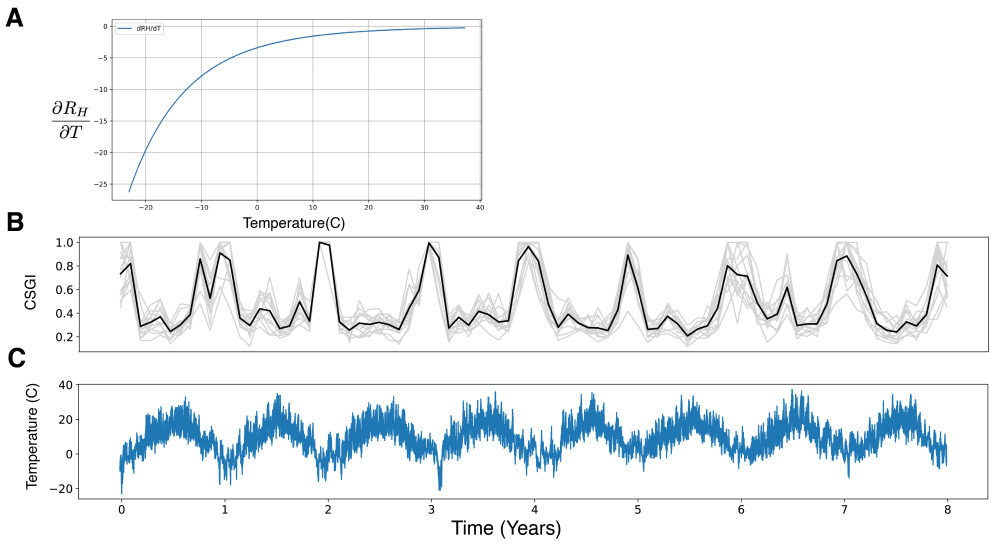


Figure 3.9: Causal interactions with relative humidity from the Jena Climate Dataset. **A**) Empirical relationship between relative humidity and air temperature (assumes $T_{dew} = 10$). Note the large negative partial derivative at low values of T. **B**) TACI predictions for causal interactions for how the other 13 variables in Table 3.1 affect relative humidity as a function of time across the eight years of the dataset (gray lines, mean trajectory is the black line). Note how causal influence peaks consistently when the temperature (**C**) is at its nadir, just as predicted by the plot in **A**.

recorded activity from a hemisphere of the monkey's brain that covered the visual, temporal, parietal, motor, prefrontal, and somatosensory cortices, sampling at 1kHz (details can be found in [122]). Data were collected during both awake and anesthetized states to examine neural activity across different consciousness levels. To generate an anesthetized state, the monkey was chair-restrained and propofol was injected intravenously. The recording sessions were structured into four distinct phases: an initial phase where the monkey is awake with eyes open, a subsequent phase where the monkey is awake but with its eyes covered, a phase where the monkey is under deep anesthesia, induced to reach a state of loss of consciousness, and a final stage where the monkey recovers from anesthesia with its eyes covered. The depth of anesthesia was assessed by monitoring the monkey's responsiveness to tactile stimulation and the presence of slow wave oscillations in the ECoG signal [122].

Previous studies analyzing these data for changes in causal interactions using

Spectral Granger Causality [122] or CCM [123], but each was only able to analyze data at the level of the four phases described in the previous paragraph (each requiring training a separate model, as well). Moreover, given the performance of Granger Causality and CCM on our synthetic data sets in the previous section, we were curious whether our results would differ qualitatively or quantitatively from these approaches. Specifically, we trained TACI on one monkey (George in [124]) with a sequence length of 50 to account for the extended autocorrelation time observed in the time series (average of 53). Approximately 53 minutes of data corresponding to the four previously outlined phases were utilized for this purpose. The training was conducted over 300 epochs or until the point of convergence. Further details of the parameters used can be found in 3.2

Finally, to compare with these previous studies, while we calculated the causal interactions between each pair of electrodes, we will present many of the results as the average result between pairs of electrodes assigned to the same region of the cortex. Here, we will be using the eight coarse-grained regions defined in [123]: the medial prefrontal cortex (mPFC), lateral prefrontal cortex (lPFC), pre-motor cortex (PMC), motor and somatosensory cortex (MSC), temporal cortex (TC), parietal cortex (PC), higher visual cortex (HVC), and lower visual cortex (LVC).

Fig. 3.10 shows time-averaged values of correlation (**A**), TACI-derived causal interactions (**B**), and Directionality (**C**), which we define as the difference in CSGI values in one direction vs. the other, for epochs of time before, during, and after anaesthetization. For correlation, we measure the average Pearson correlation coefficient between all the electrodes assigned to the various regions. Note that the diagonal terms do not necessarily have to be equal to one here, as electrodes within a region are not perfectly correlated with one another. There are only minimal changes in brain region interactions across the three time windows when measuring correlation, but large differences emerge when analyzing the data using TACI. Specifically,

we see that almost all interactions disappear during the anesthetized period, with the interactions beginning to re-emerge during the recovery period. These results differ from the results from CCM in [123], where they claimed that while some interactions decreased, others strengthened (this effect is seen in our Directionality measurements, however). Also interesting are the nearly vertical lines in Fig. 3.10B, implying that certain regions like the mPFC might be affected broadly by signals from various parts of the cortex – a finding that agrees with the commonly held notion that the mPFC's role often involves higher-level cognitive function. Again, it should be noted that only one TACI network was trained per pair of interactions across all time epochs, unlike the other methods we describe, which must find interactions separately during each measurement period.

Lastly, taking advantage of the aforementioned property of TACI, we took a finer-grained look at how interactions between a pair of regions might change with time during the experiment, specifically the mPFC and the PC. In Fig. 3.11, we show how these regions' interactions alter with time. Using our approach, we observe how the coupling slowly decays upon administration of the propofol and how it rapidly increases a few minutes into the recovery period. Also interesting is that while during the awake periods, PC consistently has a casual interaction towards mPFC, the reverse interaction has significant temporal fluctuations whose study might lead to insights into how these brain regions drive each other during cognitive tasks.

3.4 Discussion

In this chapter, we introduce a new methodology for probing time-varying causal interactions in complex dynamical systems using a novel machine learning architecture for causal inference, Temporal Autoencoders for Causal Inference (TACI), combined with a novel metric for assessing causal interactions using surrogate data. A particu-

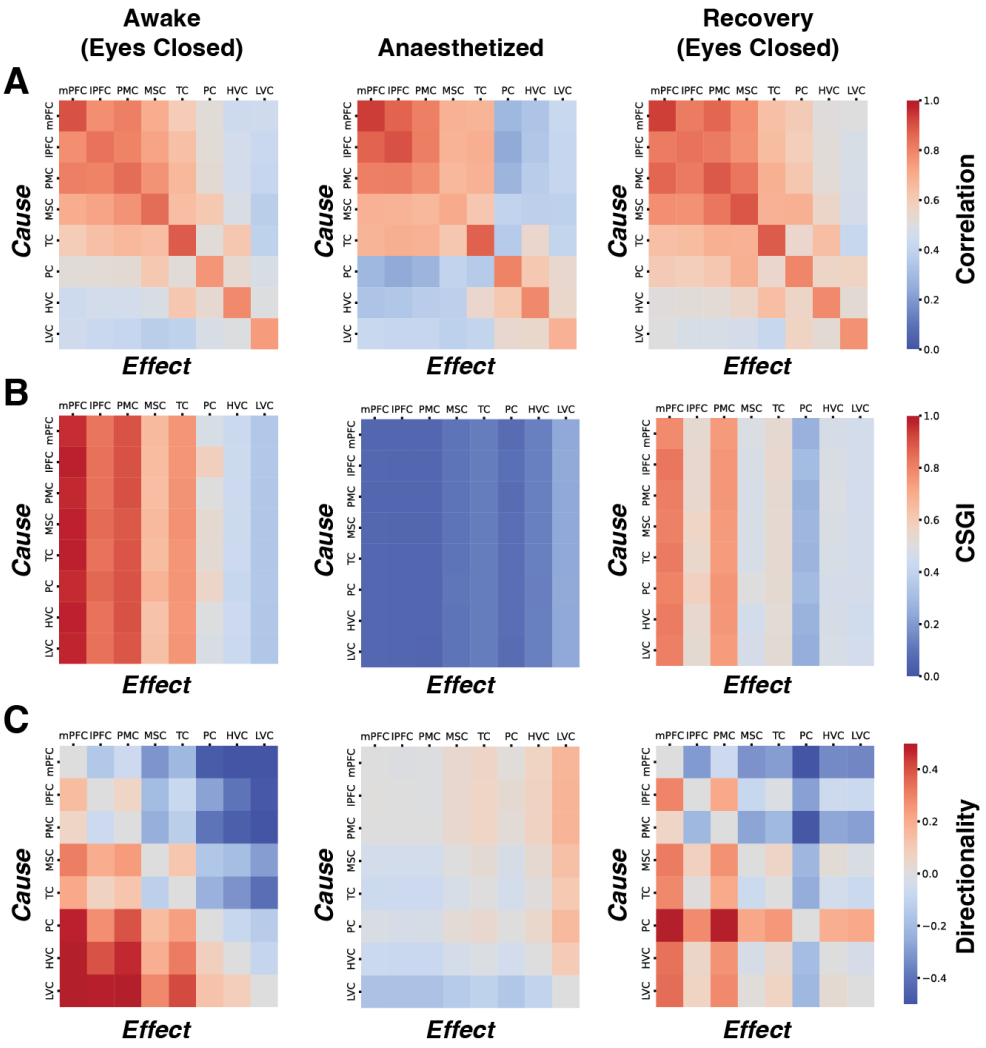


Figure 3.10: Interactions between brain regions in ECoG data. Each plot here shows the average interaction between all electrodes within each of the 8 coarse-grained regions described in the text. The left matrices are from before the anesthesia was administered, the middle matrices are from when the monkey was anesthetized, and the right plots are from the recovery period. **A** is the Pearson correlation between the signals, **B** is the TACI-derived inference of causal interaction, and **C** displays the TACI Directionality – the difference between the CSGI score in one direction minus the CSGI score in the other direction.

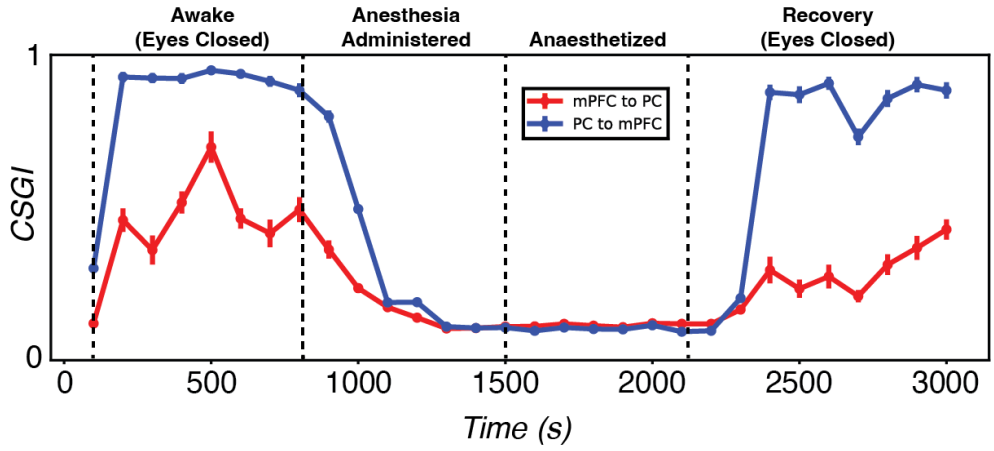


Figure 3.11: Causal interactions across time between Parietal and medial Prefrontal Cortices. Plot of the average TACI-derived interactions between PC and mPFC over the course of the anesthesia experiment. Error bars are the standard errors of the mean across all electrode interactions, and the dashed lines represent change points in the experimental protocol (labeled above the axes).

lar advantage of our approach is being able to train a single model that captures the dynamics of the time series across all points in time, allowing for time-varying interactions to be found without retraining, a computationally expensive endeavor for most artificial neural networks. We found that our method performed well compared to other methods in the field on synthetic data sets with known causal interactions, including those with time-varying couplings between variables. We also found that our method was able to identify known interactions between variables in a climate data set and was able to discover subtle temporal fluctuations in coupling in non-human primate ECoG data.

Our approach, while novel, is not without its limitations. One of the primary concerns is the extensive training time and the resource-intensive nature of the model. Implementing TACI, especially on large datasets, requires significant computational power and time. We envision that several technical improvements in the network architecture and training will allow for the method to be sped-up considerably, however. Another concern is the potential for overfitting due to TACI's considerable modeling capacity. While the framework is designed to capture the nuanced dynamics of causal

relationships over time, like most other causal network models, this method can fit data too closely if not trained properly, resulting in models that perform exceptionally well on training data but generalize poorly to unseen data. Furthermore, TACI incorporates elements of the Granger causality approach, which means it also inherits some of its problems. Granger causality assumes that the causal variable contains unique information about the future values of the effect variable, which might not always hold true in complex systems where numerous latent factors influence outcomes. Lastly, but importantly, as our approach is based solely on observational data, TACI only attempts to provide hypotheses about causal relationships between variables or to infer important relationships between variables when perturbation experiments are impossible or unethical to perform.

These limitations withstanding, however, the results presented in this chapter provide evidence that our approach will be broadly applicable to complicated data sets with time-varying causal structure, with particular promise for neural data, where we hope to build our understanding of how parts of the brain shift their interactions as behavioral states and needs alter in the world.

3.5 Materials and methods

At its core, TACI uses a two-headed autoencoder architecture implemented in a two-step process aiming to facilitate the prediction of future states and the inference of causal relationships between different time series datasets. In the first application, the two-headed autoencoder is utilized to process the original time series data, $x(t)$ and $y(t)$. The encoder segments of this autoencoder independently process $x(t)$ and $y(t)$, capturing and encoding their temporal dynamics and features into latent representations. These representations are then merged in the bottleneck, combining the distilled information from both time series into a unified latent space that en-

capsulates potential causal interactions. From this combined latent representation, the decoder works to reconstruct or predict the future trajectory of $y(t)$, shifted by a time τ. The second application involves replacing $x(t)$ with the surrogate data $x^{(s)}(t)$. This surrogate data is generated to mimic the statistical properties of $x(t)$ but is designed to break any potential causal link between $x(t)$ and $y(t)$

This two-step process is essential for figuring out how these variables are linked to one another. The model can validate the presence of a causal relationship by comparing the predictive accuracy of the decoder when using the original $x(t)$ versus the surrogate $x^{(s)}(t)$. A significant drop in accuracy with the surrogate data suggests that the original $x(t)$ contains specific information causally linked to the future states of $y(t)$.

3.5.1 Architecture

In the TACI architecture, the concept of a two-headed encoder is employed to simultaneously process two distinct time series datasets, denoted as $x(t)$ and $y(t)$. This design allows for the independent yet parallel analysis of each time series, enabling the model to capture and encode their individual characteristics and temporal dynamics before merging their representations during the bottleneck process. The input sequences are selected to be greater in length than the autocorrelation time of each variable. This ensures that the sequences capture meaningful temporal dependencies and dynamics. A GaussianNoise layer is added to enhance the model's ability to generalize and prevent overfitting.

The most important part of the encoder includes the use of a Temporal Convolutional Network (TCN) layer. Thus, capturing the long-term dependencies within each time series. This layer utilizes several key parameters: *"nb_filters"* sets the number of convolutional filters, *"kernel_size"* affects the temporal extent of each convolution, *"dilations"* allows the model to efficiently gather information across various temporal

distances. Additionally, *"Dropout"* layers are used to decrease overfitting by randomly dropping units during the training phase. Following the TCN, a *Conv1D* layer continues to process the data for each series, allowing the network to change dimensionality while preserving temporal resolution. An *AveragePooling1D* layer may then downsample the *Conv1D* layer's output by pooling across the temporal dimension. This operation reduces the sequence length, emphasizing significant features and further decreasing data dimensionality. The data is subsequently processed by a series of *Dense* layers that compress it into a dense, lower-dimensional latent representation. The size of these layers decreases in each successive layer, concentrating the information into a more compact form.

The bottleneck stage starts once the two-headed encoder has finished processing and compressing the input sequences into a lower-dimensional latent space representation. The Bottleneck merges these latent representations through an element-wise multiplication operation. By combining the representations in this manner, the model effectively captures the potential interactions and dependencies between the time series, which are essential for uncovering causal relationships.

Once the latent representations are merged in the Bottleneck, this combined representation is forwarded to the Decoder. The Decoder's task is to predict the future trajectory of the target time series. The first step in the Decoder is to progressively upscale the combined latent representation. This is achieved through a series of *Dense* layers, where each layer aims to increase the dimensionality of the data. The number and size of these layers are determined by the complexity of the data and the level of compression achieved by the Encoder. After the initial upscaling, an *UpSampling1D* layer is used to increase the sequence length to its original size, effectively reversing the pooling operation performed in the Encoder. A TCN layer is used to ensure that the reconstructed data maintains its temporal integrity and dynamics. This layer mirrors the TCN configuration in the Encoder, utilizing the same parameters for

"nb_filters", *"kernel_size"*, and *"dilations"* to capture the temporal dependencies and patterns necessary for accurate prediction. Lastly, a Dense output layer produces the final prediction of the future states of the target time series.

3.5.2 Training and Prediction

As discussed earlier, the training phase of the TACI model involves four distinct configurations of the network. Central to this phase is the use of the Mean Squared Error (MSE) as the loss function, which facilitates the optimization of predictions for future trajectories against actual observed values. The Adam optimizer [125] is employed for its adaptive learning rate capabilities. Training is done across 300 epochs to give the model enough time for the parameters to adjust and converge toward optimal solutions. The parameters controlling the batch size and data shuffling are finely tuned to balance computational efficiency and the promotion of model generalization. Callbacks such as ReduceLROnPlateau, EarlyStopping, and ModelCheckpoint are employed in this phase for optimizing the training process by adjusting learning rates, preventing overfitting, and preserving the best model state, respectively.

Surrogate data were created by initially converting the original series into the frequency domain through a Fourier transform. Subsequently, we applied random phase shifts, making sure the amplitude spectrum remained unaltered. This randomness is crucial to breaking any specific temporal dependencies present in the original series. Following this process, an inverse Fourier transform was employed to reconstruct the series back into the time domain. This step generates a new time series that mirrors the original in terms of its overall power distribution but only has random contingencies with its partner data set.

After training is completed, the model moves on to the prediction phase, where the focus shifts to evaluating the trained model. In the first step of the prediction phase, the pre-trained models are loaded, each representing a unique configuration

designed in the training phase to capture and analyze the causal dynamics between the time series datasets $x(t)$ and $y(t)$. At the same time, the full original dataset is divided into sequences with the same length and structure as the models were trained on. The prediction process occurs over defined rolling windows to allow for a temporal exploration of the dataset, enabling the models to make predictions for future states of the time series within each window. The models' accuracy in forecasting future time series states is quantitatively evaluated for each rolling window using the R^2 metric. To enhance the reliability and confidence of these assessments, 100 bootstrap samples are generated for each window. The causal inference for each rolling window can be determined using the CGSI Eq. 3.6. Through this calculation, the model not only quantifies the strength and direction of the causal relationship but also shows its variation over time, providing a dynamic and temporal perspective on causal inference.

For each interval, a bootstrap strategy is implemented. This strategy involves creating a set number of surrogate samples by randomly resampling within the interval. These samples are then used to evaluate the model's predictions, which are generated under two conditions: one using the actual interactions between the time series and another using the surrogate data. By employing Equation 3.6, it's possible to derive scores from which we compute both the mean and standard deviation. These computations provide insight into the average performance and variability of the model's predictions across the bootstrap samples. The utilization of bootstrap methods significantly enhances the analytical depth by ensuring that the derived error bars and confidence intervals are supported by a solid statistical foundation. These statistics play a vital role in establishing the error bars in the plotted figures. By repeating this procedure across all intervals, the method provides a comprehensive view of how model performance fluctuates over time and under different conditions.

3.5.3 TACI Network Parameters

Parameter	Description	Value
nb_filters	Number of filters in TCN layers.	32
kernel_size	Size of the kernel in TCN layers.	32
dilations	Dilation rates for TCN layers.	[1, 2, 4, ..., 32]
nb_stacks	Number of stacked TCN layers.	1
ts_dimension	Dimensionality of the time series.	1
dropout_rate_tcn	Dropout rate for TCN layers.	[0.0, . . . , 0.5]
dropout_rate_hidden	Dropout rate for hidden layers.	[0.0, . . . , 0.5]
conv_kernel_init	Kernel initializer for convolutional layers.	'he_normal'
latent_sample_rate	Downsampling rate in the latent space.	2
act_funct	Activation function used in layers.	'elu'
epochs	Number of training epochs.	300
batch_size	Batch size for training.	512
shuffle	Whether to shuffle the data during training.	[True or False]
scaling_method	Method used for scaling the input data.	Z score
loss_funct	Loss function used for training.	'mse'
noise	Standard deviation of Gaussian noise added.	[0.0, . . . , 0.5]
window_len	Size of the rolling window for predictions.	value
seq_length	Length of sequences used for training/prediction.	[10, . . . , 100]
lag	Lag between $x(t)$ and $y(t)$ for prediction.	[10, . . . , 100]

Table 3.2: Parameters used in the TACI model training and prediction phases (ranges indicate the parameter range used across the examples in this chapter)

Chapter 4

Building emergent representation of behavioral states using dynamical models

4.1 Introduction

Behavior is inherently dynamic, requiring a vast range of intricate coordinated movements. These movements are not completely random but are instead a precise combination of neural and muscular control in response to a multitude of internal and external stimuli. These movements are essential for survival and reproduction, enabling the animals to find sources of food, attract mates, and escape potential predators.

When experimentally probing an animal's repertoire of actions, the most common approach is to isolate *stereotyped behaviors* [31]. These often-repeated and consistent behaviors are executed in response to specific internal states. Isolating these stereotyped behavioral states has been of intense interest in recent years [27, 28, 29, 30]. These methods rely on state-of-the-art algorithms to recognize, classify, and measure the stereotypical actions displayed by animals in their everyday lives.

A common issue with these methods, though, is that they typically only can be measured at a single time scale [28]. Behavior spans across a multitude of scales, however. For instance, a behavior can be a split-second reflex action, but there are also other behaviors that take place over minutes or hours, such as the mating dances of certain birds (peacocks, cranes, etc.) or the hunting patterns of predators. This scale is often chosen based on the results either from arbitrary experimental observations or from fitting a Markov model to the system [126], but creating a representation of stereotyped behaviors that spans multiple time scales remains an unresolved problem.

In this Chapter, we introduce a novel approach for measuring the entire behavioral repertoire of an animal's behavior that does not rely on a single time scale. This method uses Recurrent Neural Networks (RNNs) to create a dynamical system model of postural time series and then finds the fixed points of that model. These fixed points are what we will use to build our multi-time-scale representation. We applied this method to 3D kinematic data from freely behaving rats that captures the movements of a given rat's head, trunk, and limbs over extended periods, typically spanning week-long timescales. By identifying these primitives, we can begin to piece together how complex behaviors may be constructed across multiple time scales.

4.1.1 Methods

A foundational concept in the study of dynamical systems is the identification of fixed points within the phase space. Fixed points are characterized by their lack of motion in phase space and because the dynamics near them can be approximated as linear, making them easier to analyze and understand. These points are pivotal for understanding the system's behavior, as they often represent stable states or attractors that the system gravitates towards under certain conditions. Our goal is to build a dynamical model of an animal's behavior and to find the fixed points resulting from that model.

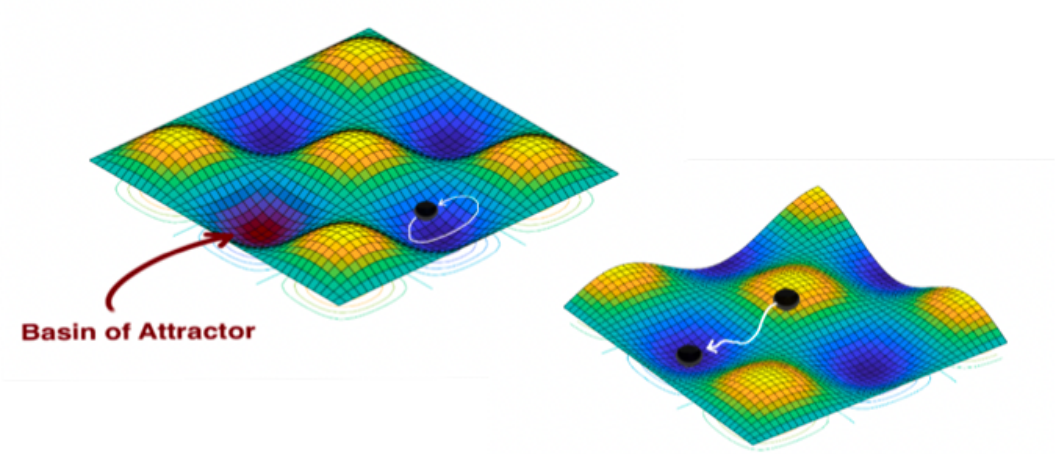


Figure 4.1: Multi Basin Behavioral Landscape. Each basin signifies a stable behavioral state toward which the system naturally converges. The barriers separating these basins represent the thresholds for transitions, symbolizing shifts in behavior. The white arrow is a trajectory suggesting a potential behavioral shift from one basin to another

Because of the complexity of the system, we relied on artificial neural networks to approximate the dynamical system that we will then analyze. Specifically, we trained a sequence-to-sequence Recurrent Neural Network (RNN) on the processed data (see Section 2.3.3). Viewing the RNN as a nonlinear dynamical system offers a rich framework for analysis (more in Section 4.1.2). This perspective allows for the examination of the internal states of an RNN across different regions of phase space. The idea of fixed points in the internal states of the RNN provides a compelling framework for defining behavioral states. The interactions among these fixed points create a complex landscape of behavioral possibilities characterized by a basin-like structure (more in Section 2.3.1). In this landscape, certain sets of states form the "bottom" of basins, acting as attractors that other states gravitate towards. The way the system is organized suggests that transitions between these basins, as shown in Figure 4.1, may be used to explain the behavior of the system. Each basin corresponds to a distinct behavior, and transitions between them represent changes in behavior. If we compare the dynamical system of internal states to a particle moving in a multi-

well landscape, we can gain insights into how behavioral transitions occur. Each well or basin in the landscape corresponds to a stable state, with the barriers between basins representing the thresholds that must be overcome for transitions to take place.

4.1.2 Recurrent Neural Networks Fixed Points

Fixed points in the context of Recurrent Neural Networks (a more broad definition in Section 2.3.3) refer to states of the network where the output of the RNN for a given input is equal to the input of the RNN. In other words, when the network reaches a fixed point, the state of the network remains constant despite the continuous application of the network dynamics. For simple networks, it is possible to analytically solve for fixed points. However, for larger and more complex networks like RNNs, researchers often have to rely on iterative numerical techniques to approximate these points. Once the fixed points have been identified, their stability can be analyzed numerically by examining the eigenvalues of the Jacobian matrix of the network's dynamics at that point.

Considering an RNN as a high-dimensional dynamical system, its behavior at any given time step can be described by the state update equation:

$$h_t = f(W \cdot h_{t-1} + V \cdot x_t + b), \tag{4.1}$$

where h_t is the hidden state at time t, x_t is the input, W and V are weight matrices for the recurrent and input connections, respectively, b is a bias term, and f is a nonlinear activation function. Now, if we simplify this system to a sequence of length 3 as in [33]. The evolution of the network's state can be iteratively computed as

follows, starting from an initial state h_0:

$$
\begin{aligned}
h_0 &= h_0, \\
h_1 &= f(Wh_0 + Vx_0), \\
h_2 &= f(Wh_1 + Vx_1), \\
h_3 &= f(Wh_2 + Vx_2).
\end{aligned}
$$

This iterative process suggests a fixed-point formulation where the sequence of states $\vec{h}$ can be expressed as a function of itself and the sequence of inputs $\vec{x}$, leading to a fixed-point equation:

$$\vec{h} = \vec{f}(\vec{W} \cdot \vec{h} + \vec{V} \cdot \vec{x}). \tag{4.2}$$

The Banach Fixed Point Theorem, or the Contraction Mapping Theorem, provides conditions under which a function f on a complete metric space will have a unique fixed point towards which iteratively applying f will converge [127]. The theorem not only guarantees convergence to a fixed point but also states the following iterative process $\vec{h}_{n+1} = f(\vec{h}_n; \vec{x})$ will converge exponentially fast to the fixed-point of $\vec{h} = f(\vec{h}; \vec{x})$. This means that the distance between the state $\vec{h}_n$ at iteration n and the fixed point decreases exponentially with n. In the context of RNNs, the equation

$$\vec{h}_{n+1} = \vec{f}(\vec{W} \cdot \vec{h}_n + \vec{V} \cdot \vec{x}), \tag{4.3}$$

describes the corresponding iterative process where $\vec{h}_n$ represents the state of the network at iteration n.

This process guarantees that the system's dynamics are stable and predictable. In other words, the convergence to a steady state does not depend on the network's initial conditions. Moreover, it speeds up the analysis by allowing parallel computations of all historical states.

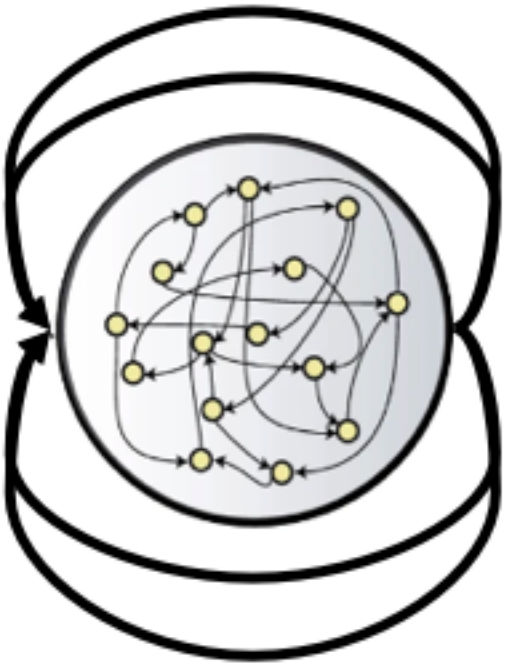


Figure 4.2: RNN iterative self-feeding mechanism. The RNN generates an output based on an initial input sequence. This output, representative of the network's prediction for the next state, is then recursively fed back as the new input in a continuous loop. This self-feeding cycle is repeated iteratively until subsequent inputs no longer result in significant changes to the internal states.

Consequently, once the network has been trained, the network's output can be used as the initial state of the iterative process for further analysis, particularly for deriving the fixed points of the network. This approach involves setting the network's output as the starting point and then iteratively applying the network's dynamics without any external input to see how the internal state evolves according to the equation:

$$\vec{h}_{n+1} = \vec{f}(\vec{W} \cdot \vec{h}_n + \vec{V} \cdot \vec{x}_n), \tag{4.4}$$

where $\vec{x}_{(n)}$ is the output of network from the previous state.

4.2 Data

The data we used for this analysis are 3D motion-tracked kinematic data from freely behaving rats [128]. Here, researchers built a specialized rodent motion capture studio to facilitate precise tracking. This setup included a two-foot-diameter plexiglass arena surrounded by 12 motion capture cameras strategically positioned to minimize occlusions and ensure comprehensive coverage of the rat's movements. The arena was outfitted with bedding, various objects, and a lever for operant conditioning, promoting a range of natural behaviors.

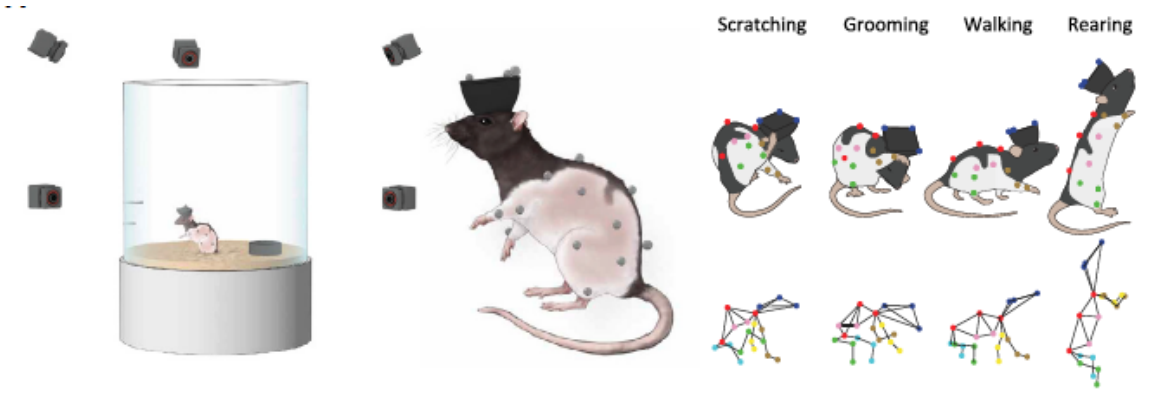


Figure 4.3: Schematic depictions of the rats' arena and attached markers. Adapted from the study by Marshall *et al.* on continuous whole-body 3D kinematic recordings across the rodent behavioral repertoire [128].

Despite the advanced setup, vision-based tracking systems, including this one, are susceptible to marker dropouts caused by the animal's self-movement or environmental factors, leading to temporary loss of tracking data. Notably, these dropouts, especially affecting the forelimb and hindlimb markers, were predominantly short-lived, lasting around 20 milliseconds. To counteract these gaps, the researchers employed standard interpolation techniques, leveraging the temporal history of each marker's position. This method allowed for the accurate reconstruction of the markers' positions during brief occlusions, ensuring the continuity and reliability of the kinematic data.

The continuous kinematic recordings obtained through this sophisticated setup enabled the creation of a definitive reference dataset of rat behavior. This dataset catalogs an extensive array of movements performed by the rat over week-long periods, providing an unprecedented level of detail and insight into the behavioral repertoire of freely behaving animals.

4.2.1 Analysis Pipeline

The general framework of our analysis is described in Figure 4.4. The initial phase involves decomposing the collected 3D kinematic data into postural time series. This decomposition allows us to isolate specific postural modes and their temporal evolution, providing a granular view of the rats' movements.

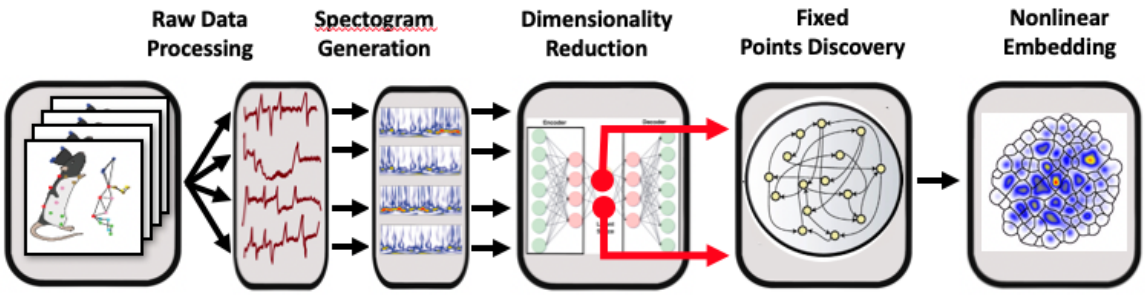


Figure 4.4: Overview of the data analysis pipeline.

These time series are then transformed into wavelet spectrograms, offering a spatio-temporal representation of the rats' dynamics. The spectrograms serve as a powerful tool for visualizing the frequency and intensity of postural changes over time, effectively capturing the essence of the rats' movements in both space and time.

The next step involves the use of an autoencoder as a dimensionality reduction tool. We use the wavelet spectrograms as input into the autoencoder. The autoencoder is trained to compress the high-dimensional spectrograms into a lower-dimensional latent space, extracting the most salient features that characterize the original data. This process results in a condensed representation that retains the most important information about the rats' behavioral dynamics while reducing the complexity of the data at the same time.

Once the reduced-dimensional spectrogram data has been obtained from the autoencoder, we proceed to train a Long Short-Term Memory (LSTM) network. This network is tasked with predicting future values of the rats' movements based on the learned representations. This predictive modeling step is crucial for understanding the temporal patterns and potential future states of rat behavior. After training and validating the LSTM network, we shift our focus to the analysis of fixed and slow points within the network. These points are of particular interest as they can reveal stable states or attractors in the rats' behavioral dynamics.

We continue by embedding the candidate fixed points into a two-dimensional space via t-SNE. This process not only simplifies the visualization but also reveals the basins of attraction within the behavioral dynamics of rats. To delve deeper into the structure of these basins of attraction, we proceed to estimate the probability distribution over the two-dimensional space generated by t-SNE. This step is important for identifying the density and distribution of fixed points within the embedded space. Lastly, we estimate the probability distribution over this dimensional space and discover peaks in the distribution.

Postural decomposition

We use data corresponding to 3D movement kinematics encompassing the entire behavioral repertoire of rats. This comprehensive dataset documents a wide variety of movements performed by different rats over prolonged periods, specifically across week-long timescales. The dataset utilized in this study is derived from 19 strategically placed markers on the animals, specifically targeting the head, trunk, forelimbs, and hindlimbs. This comprehensive marker setup ensures a detailed capture of the full range of motion exhibited by the animals.

We carefully created a subset of this data, selecting only those segments exceeding a certain length threshold—those comprising more than 10,000 frames, equivalent to approximately 27 minutes of continuous data. To address the memory constraints of our computational capabilities, the dataset was downsampled from its original recording rate of 300 Hz to a more manageable 60 Hz. The final dataset utilized for analysis was carefully screened for correctness and clarity during this selection process to make sure it was free of noise, artifacts, and tracking errors that would have compromised the study's findings. In order to get accurate and dependable insights into the intricate dynamics of the rats' movement and behavior, it was imperative to prioritize the selection of clean data segments.

Following the identification of a suitable dataset, we applied an egocentric technique to the selected data. This process was performed on the x and y coordinates of each rat's body part segment, while the z coordinate was left unmodified. This decision was made to accurately capture and preserve the vertical movements of the rats, such as instances where they rise on their hind legs, a behavior often referred to as "rearing." Egocentering involved repositioning the data so that the subject rat was consistently positioned at the center of the coordinate system for each analyzed segment. By centering the subject rat within the coordinate system, we eliminated variations in the horizontal orientation.

Spectrogram Generation

Creating a spectrogram for postural dynamics analysis provides a more comprehensive approach to understanding behavior beyond the limitations of instantaneous time series values. Traditional studies in behavior have sought to identify motifs to uncover patterns and trends. However, these approaches encounter significant challenges, such as issues with temporal alignment and relative phasing between the component time series [27].

As a solution to these problems, the spectrogram representation emerges as a powerful alternative. This method focuses on measuring the power at a specific frequency f associated with each postural mode, $y_k(t)$, within a time window centered around a moment in time, $S(k, f; t)$ (details of these calculations are shown in Section2.3.4). The amplitudes for each postural mode are computed using the Morlet continuous wavelet transform, which allows the spectrogram to capture the core of postural dynamics over a variety of time scales. In contrast to a spectrogram, which offers a fixed resolution across all frequencies, wavelets provide a multi-resolution time–frequency trade-off that allows for a more detailed examination of phenomena occurring at various time scales.

In this particular case, the spectrogram contains 30 frequency channels that are dyadically spaced between 1 Hz and 60 Hz. This approach ensures a comprehensive coverage of the frequency spectrum, allowing for a detailed and dynamic representation of postural modes. Importantly, the wavelet transform amplitudes provide insight into the temporal and frequency distribution of postural dynamics, offering a clearer understanding of the underlying behaviors.

Latent Dynamics

The use of a deep autoencoder for dimensionality reduction of high-dimensional data is a generalization of traditional linear methods like PCA. However, contrary to these

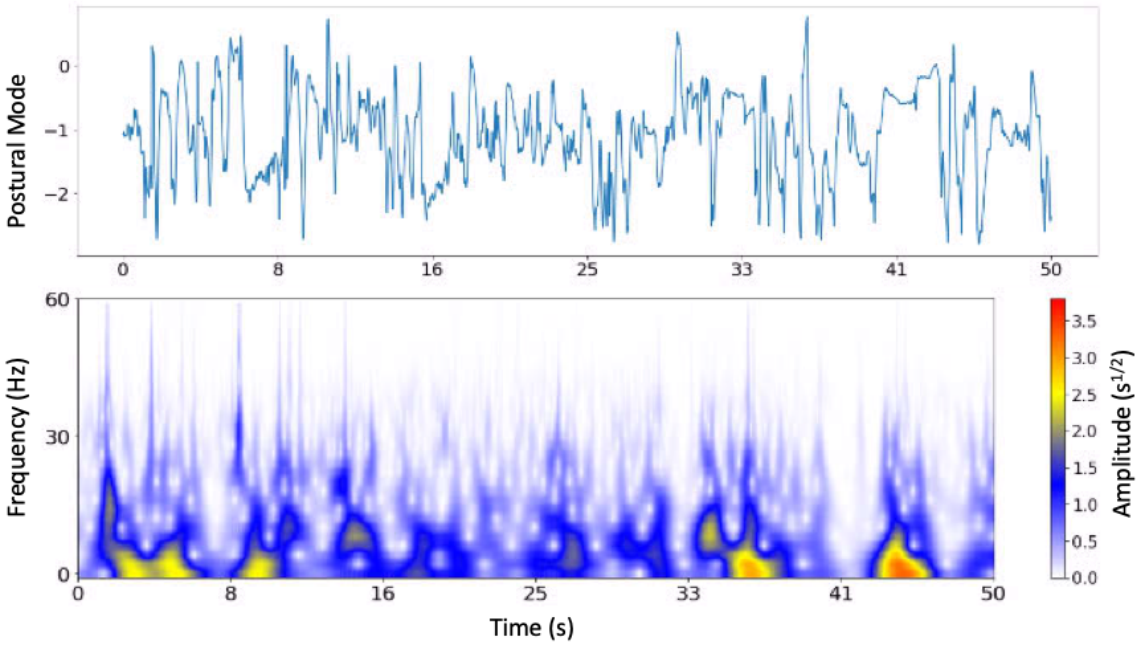


Figure 4.5: Example wavelet transform of postural data. Top: Typical time series from one of the three coordinates of a specific body part. Bottom: Its corresponding wavelet transform

methods, autoencoders address both linear and nonlinear correlations. This ability is the reason why we chose it: to better identify and represent the underlying structure of the data. More details can be found in Section 2.3.5.

The deep autoencoder employed in this study is constructed exclusively with dense layers. These types of layers are effective at learning complex patterns and relationships present in the data. The autoencoder's architecture is designed to compress the high-dimensional input spectrogram into a more compact representation in the latent space before reconstructing it back to its original form.

In this analysis, the spectrogram utilized contains 30 frequency channels, dyadically spaced between 1 Hz and 60 Hz. This frequency distribution is applied to each coordinate of every body part marked for tracking, with a total of 19 different body markers employed. Consequently, this results in a substantial input sequence for the autoencoder, with a length of 1710.

The choice of using wavelet transform conditions the normalization method we

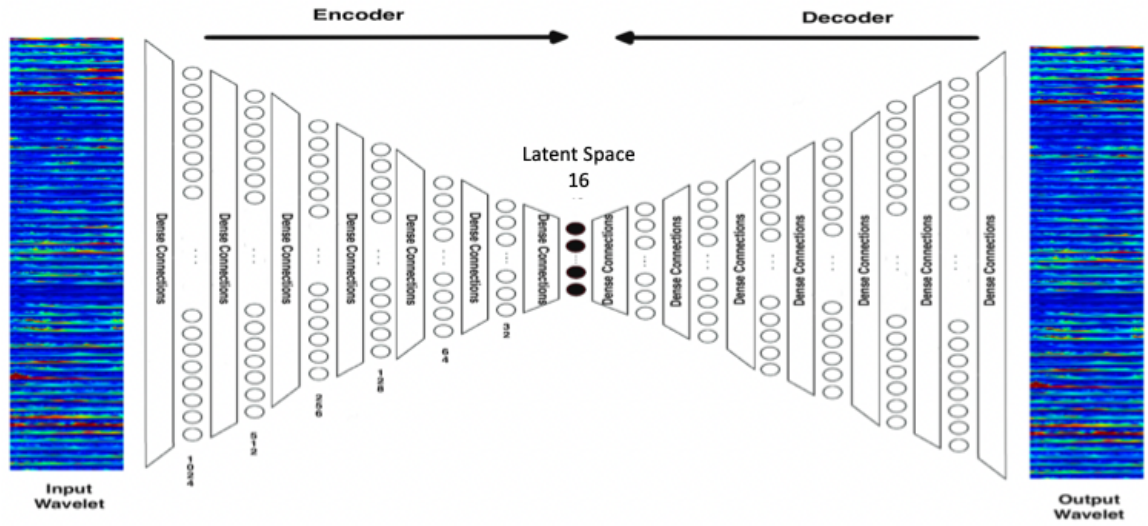


Figure 4.6: Diagram of the autoencoder architecture utilized for the dimensionality reduction of wavelet-transformed behavioral data. The input layer contains the high-dimensional data derived from wavelet transforms of behavioral sequences. The latent space conserves the critical information of the input data in a reduced form while discarding redundancies. Following the compression, the decoder reconstructs the original input data from the reduced latent representation.

used for these sequences. Traditional normalization methods, such as norm are highly sensitive to amplitude modulations and do not provide a reliable measure of the spectral differences. However, since each sequence is inherently positive and semi-definite, we can transform it into a probability distribution function (PDF) over all mode-frequency channels at a given time point so that each sequence would sum up to 1.

We designed this deep autoencoder to manage the complexity and high dimensionality of these input sequences. The initial layer of the autoencoder is densely populated with 1024 neurons, establishing a robust foundation for capturing the intricate patterns within the data. Subsequent layers follow a systematic reduction in complexity, with each layer halving the number of neurons in a power of 2 sequences. This progressive reduction continues until the network converges to the latent space, which is deliberately constrained to 16 dimensions.

For activation functions, we employed “ELU” for the intermediate layers and “soft-

max" for the output layer. The choice of "softmax" for the output layer is necessary to convert the output of the autoencoder into a probability distribution. This function is able to transform the output layer's raw prediction scores into probabilities by exponentiating and normalizing each score, ensuring that the sum of probabilities equals one. This allows us to compare the results from our autoencoder to the data normalized as PDFs. Additionally, we used the Kullback-Leibler divergence ("kld") as our loss function. This algorithm is particularly effective in measuring how different one probability distribution is from another. Given that each input sequence is a PDF, it makes sense to use a function that can quantify how similar or different our reconstruction sequence is relative to the original distributions.

Several configurations of the autoencoder were rigorously tested to identify the optimal structure. The criterion for selecting the final configuration was based on the autoencoder's ability to accurately reconstruct the original input data from the compressed latent representation. This specific configuration achieved an R-squared (R^2) value of 0.997 in the reconstruction. This metric indicates a higher degree of effectiveness in the reconstructed spectrogram than the original, suggesting that the autoencoder successfully captures the underlying structure and dynamics of the data within the reduced-dimensional latent space. An R^2 value so close to 1 signifies that the model accounts for nearly all the variance in the original data.

To further validate the effectiveness of the autoencoder in capturing the essential dynamics of the spectrogram data, a secondary test was performed using test-set data. This test-set data was created by randomly selecting sequences from the original dataset, ensuring that the test set is representative of the broader data while at the same time being distinct enough to serve as a valid test for the model's generalization capabilities. The selected testing data was then passed through the trained autoencoder. The autoencoder, having learned to compress and reconstruct the data based on its training, applied this learned transformation to the testing data. This

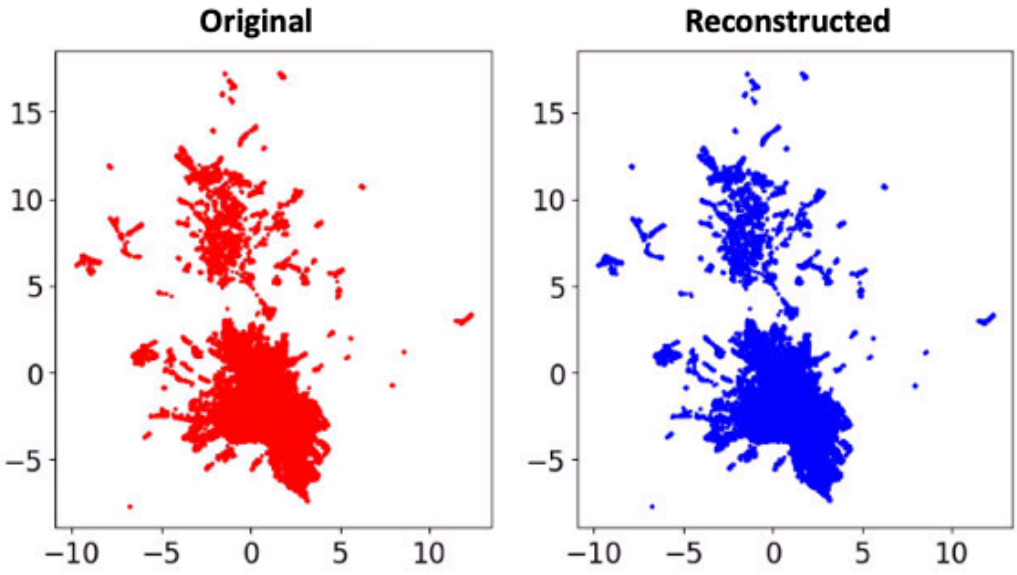


Figure 4.7: Validation of Autoencoder Reconstruction with Test-Set Data and t-SNE Dimensionality Reduction. Test data is processed through the trained autoencoder, which applies its learned compression and reconstruction capabilities to generate a reconstructed version of the test data. Both the original test data and the reconstructed version are subjected to t-SNE for dimensionality reduction to two dimensions.

resulted in a reconstructed version of the testing data.

In addition, we used t-SNE on both the original testing data and the reconstructed testing data for dimensionality reduction to two dimensions. This step was done to highlight any discrepancies in the autoencoder's reconstruction ability. This application yielded two-dimensional plots that did not reveal any apparent differences.

Fixed and Slow Transition States

LSTM networks are especially useful for sequence prediction tasks because of their ability to effectively learn long-term temporal relationships. Therefore, after successfully transforming the original spectrogram to the dimensions of the latent space via the autoencoder, we can use this reduced representation for predictive modeling. In this process, we employ a deep, stateful, one-layer LSTM sequence-to-sequence model (more information in Section 2.3.3). This model is specifically designed to forecast the

subsequent time instance (t+1) based on the current and past information encoded in the latent space representation.

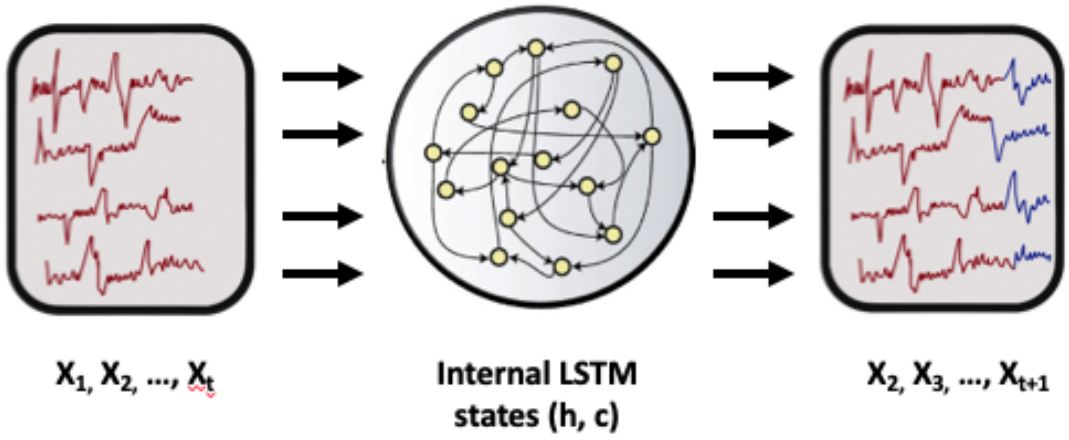


Figure 4.8: Schematic representation of LSTM neural network processing time series data. The diagram illustrates the flow of information from the input sequence $X_1, X_2, \ldots, X_t$ through the internal LSTM cell states (h, c), and ultimately projecting the transformed data to the output sequence $X_2, X_3, \ldots, X_{t+1}$. The middle network structure symbolizes the complex interactions within the LSTM cell that contribute to the network's ability to capture temporal dependencies. The four arrows symbolize the network's ability to process a batch of sequences simultaneously

The training process involves feeding the LSTM model with sequences derived from the latent space representation of the spectrogram data. The model learns to map these sequences to their subsequent time instances, iteratively adjusting its weights to minimize the prediction error. The LSTM layer is configured with 256 neurons. This size is chosen to provide the model with sufficient capacity to capture and model the complex temporal dependencies.

The Exponential Linear Unit (ELU) activation function is selected for its ability to handle the vanishing gradient problem, which is a common problem seen when training neural networks where the gradients become too small for effective learning. This function allows for small negative values when the input is less than zero, thus improving the gradient flow during backpropagation. Unlike the autoencoder that uses input and output sequences that are normalized to make a PDF, the sequences that are fed to the LSTM are extracted from the latent space. These representa-

tions are not probabilistic distributions. Instead, they are numerical and continuous. These sequences need a loss function that is able to reliably measure the difference between the output of the model and the original data. Therefore, the loss function we employed is the Mean Square Error (MSE). This loss is particularly useful in this type of regression problem because it emphasizes larger errors. For the optimizer, we use 'Adam' since it is highly efficient in handling sparse gradients. Moreover, its ability to adjust the learning rate dynamically makes it particularly useful for this type of model with large and varying sequences of data.

The choice of a "stateful" configuration of the LSTM layer is critical for retrieving fixed points from the network. Statefulness guarantees that the model can retain its state (cell states and hidden states) across batches. This property allows the model to maintain continuity in its internal state across different sequences.

After training the LSTM model to predict future instances based on the reduced representation of the spectrogram data, the next phase of the analysis focuses on identifying fixed and slow points within the network's dynamics. More information about this can be found in Section 4.1.2.

The first step in this process involves externally driving the network to a specific state. This is achieved by feeding the LSTM with consecutive input sequences from the dataset. The idea is to simulate a continuous flow of data that takes the network's internal states (cell states and hidden states) toward a targeted region in the state space. This process acts as a warm-up, transitioning the states from their initial inert condition to a more dynamic state that anticipates the next sequence.

Following the external driving phase, the analysis transitions to a self-driving step. In this phase, the output of the network is recursively fed back as input. This self-feeding loop continues iteratively, with the network's output at each time step serving as the input for the next step. The iterative nature of this process allows the network's internal states to evolve autonomously based on its learned dynamics.

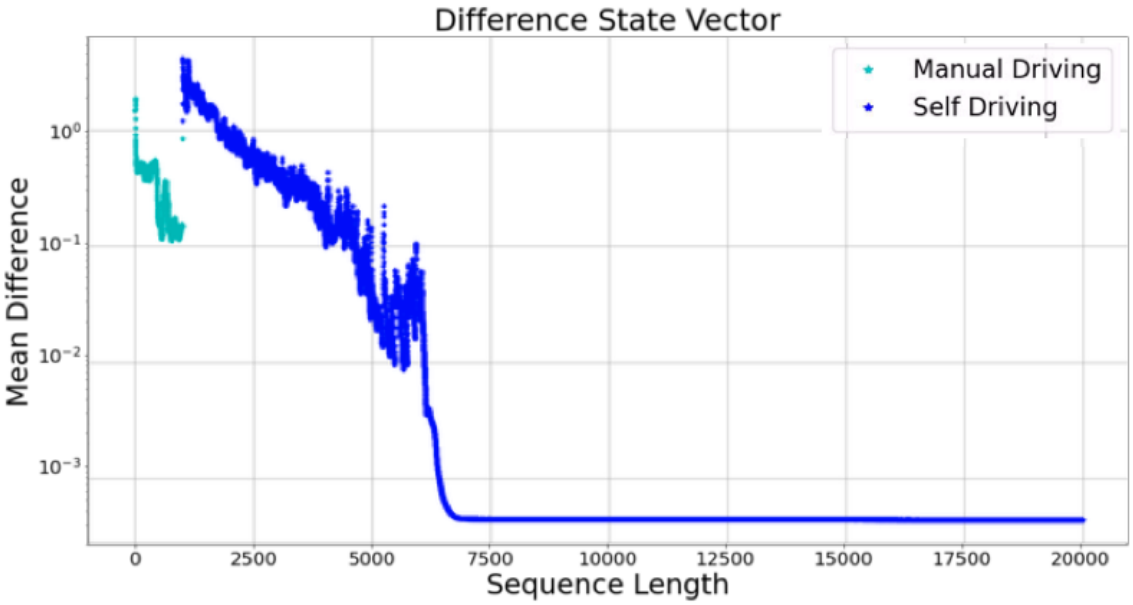


Figure 4.9: Dynamics of State Vector Evolution through External and Self-Driving Phases. During the external driving phase (light blue), the network is fed with sequences from the original dataset, simulating a natural progression of inputs that drive the network's internal states towards a specific region in the state space. In the self-driving phase, indicated by the shift to dark blue, the network begins to operate in a closed-loop manner until the network's internal states stabilize and converge towards fixed points.

The main objective during the self-driving phase is to reach the convergence of the network's hidden states (cell states and hidden states). By converge, we mean that the network reaches a state where its internal dynamics stabilize. As we can see in Figure 4.9, the difference of the state vector (cell states, hidden states, output) between successive iterations keeps decreasing until it converges. This state of convergence is indicative of the network encountering a fixed point or a slow point within its state space.

In this context, we define fixed points as the states where the network's internal dynamics come to a stop, i.e., the hidden states no longer change between successive iterations. Similarly, we can define slow points as the states characterized by a significant reduction in the rate of change of the network's hidden states without reaching a complete stop. These points suggest regions in the state space where the network's dynamics decelerate, potentially indicating transitional states.

The convergence behavior observed in the networks is expected due to their dissipative nature. This dissipative property manifests as the network's internal states (e.g., cell states and hidden states) converge towards fixed points or slow points during the self-driving phase of analysis. In physical systems, dissipation refers to the process by which energy is transformed and gradually lost from the system, typically as heat. In neural networks, dissipation can be thought of as the loss of information entropy or the reduction of uncertainty within the network's internal states as it processes data. Dissipative systems naturally evolve towards configurations that minimize their energy or, in the case of neural networks, configurations that represent stable solutions given the learned parameters and dynamics. This is analogous to physical systems settling into states of lowest potential energy.

4.3 Results

4.3.1 Embedded Space Dynamics

The analysis of the candidate fixed and slow points derived from the LSTM network faces the challenge of high dimensionality. Each fixed point has dimensions equivalent to the sum of the cell states, hidden states, and output states, making the analysis and visualization cumbersome. To address this issue, a dimensionality reduction step is needed.

The first step in simplifying the analysis involves embedding the high-dimensional candidate points into a two-dimensional space. t-SNE is a particularly efficient tool for this purpose due to its ability to maintain the local structure of high-dimensional data in a lower-dimensional space [97]. The embedding process uncovers the basins of attraction of the behavioral dynamics modeled. These basins show stable behavioral patterns or modes by representing areas in the state space where the system's states tend to converge. The next step is to estimate the probability distribution across the two-dimensional space formed by t-SNE in order to get more insight into the structure of these basins of attraction and the characteristics of the fixed points inside them [27]. More details can be found in 2.3.6.

Estimating this probability distribution is crucial for several reasons. It allows for the identification of how densely packed and distributed the fixed points are within the embedded space. Areas of higher density can indicate more prevalent or stable behavioral states. Moreover, we can identify peaks and valleys by analyzing the probability distribution. Peaks usually correspond to areas of high density in the two-dimensional space. These peaks are not arbitrary; they represent clusters of fixed points that exhibit similar characteristics, effectively grouping stable states that share common behavioral attributes. The identification of these peaks and the corresponding clusters provides profound insights into the predominant stable states within the

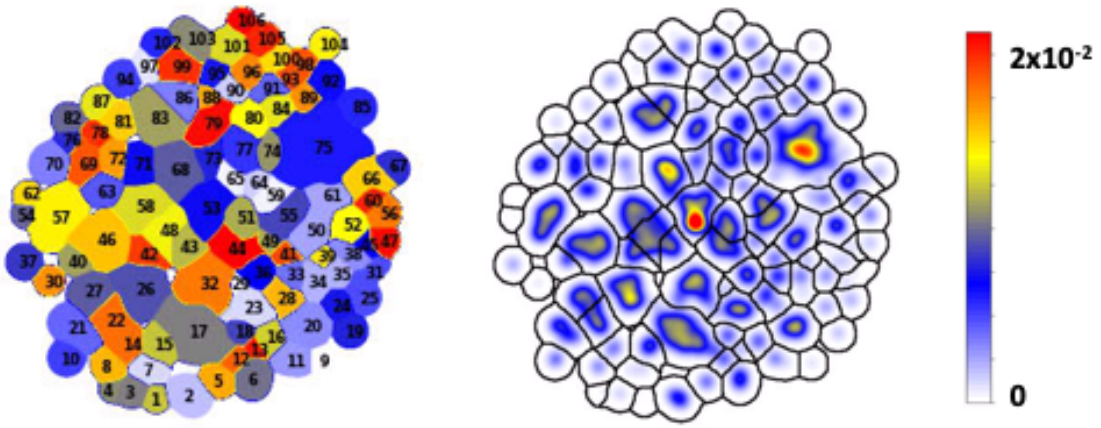


Figure 4.10: t-SNE embeddings of the model fixed points. The right plot is a probability density map of the points, and the left plot is a watershed transform of the inverse of the probability distribution

rats' behavioral dynamics. It highlights the existence of distinct modes of behavior that the rats are more likely to adopt, shedding light on the natural tendencies and preferences in their movement patterns.

4.3.2 Transition Matrices

In this Chapter, we delineate distinct behavioral states based on the density peaks in the embedded space. These peaks, or pauses at the peaks, are considered the lowest level of description of behavioral organization, representing the most stable and recurrent states within the behavioral dynamics. The pauses at these density peaks are interpreted as discrete states of behavior, each corresponding to a specific behavior the rats are more likely to exhibit. Consequently, we investigate the pattern of transitions among these states over time. This involves tracking how the system moves from one state to another, revealing the underlying structure and rules governing behavioral changes. We have a description of behavior represented as a discrete variable $S(n)$, which can assume one of $N = 106$ different values at each discrete time step n

The behavioral transition matrix is defined as:

$$[\mathbf{T}(\tau)]_{i,j} \equiv p(S(n+\tau) = i | S(n) = j), \tag{4.5}$$

which describes the probability that the rats will transition from state j to state i after a specified number of τ transition steps [28]. The matrix $\mathbf{T}(\tau = 1)$ focuses on immediate transitions, capturing the rats' behavior at its most fundamental level (Fig. 4.11 left). The predominance of self-transitions has profound consequences:

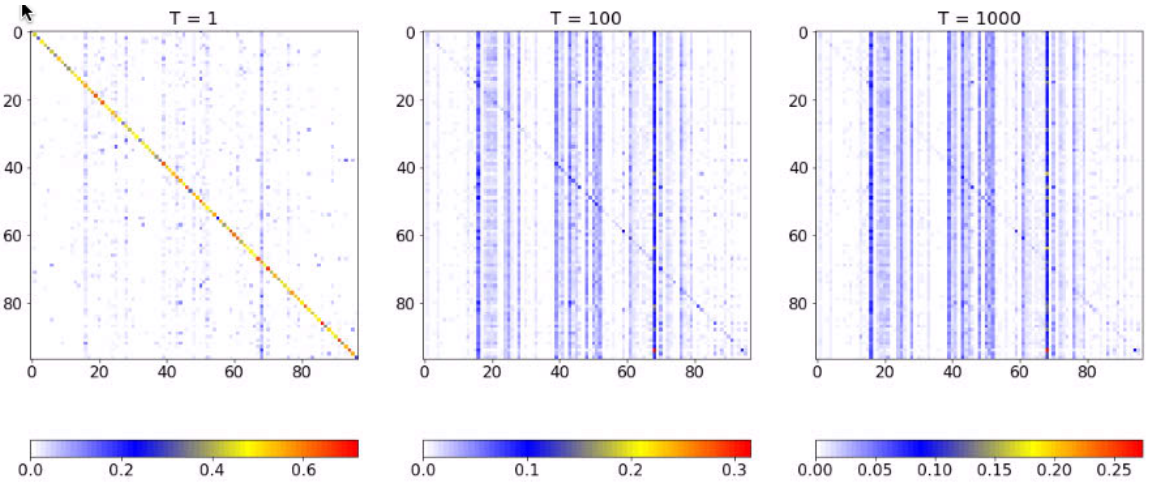


Figure 4.11: Transition Matrices.

First, it indicates that the system's states have a high degree of stability. This stability suggests that once the system enters a particular state, it is likely to remain in that state for the next time step. This persistence can be indicative of stable behavioral patterns or attractors within the system's dynamics. The system tends to remain in its current state (or basin of attraction) like a particle resting at the bottom of a well. The deeper the well, the more stable the state, and the higher the probability of self-transitions, as it becomes less likely for the system (or particle) to spontaneously move to a different state (or well).

Second, there is a high degree of inertia within the system. Changes in state are less frequent than the maintenance of the current state. This inertia is due to the

energy costs associated with changing states. In other words, the barriers between basins in the landscape space represent the thresholds that must be overcome for the system to transition from one stable state (or basin of attraction) to another. These barriers can be thought of as the off-diagonal elements in the transition matrix that are not self-transitions. A high prevalence of self-transitions implies that the barriers between different basins are significant, making transitions to other states less likely.

As the time scale τ increases (Fig. 4.11 middle to the right), it is anticipated that the structure within the distribution of transitions will deteriorate. This is a direct consequence of the decreasing predictability of future states as the prediction horizon extends. The farther out the predictions are made, the more uncertain they become.

4.3.3 Predictability and Hierarchy

Understanding that the system gravitates towards certain stable states and tends to persist in these states sets the stage for a more structured analysis of behavioral transitions and patterns. Thus, we can group these fixed points or stable states into clusters that preserve information about future actions based on the current state. This introduces a method for analyzing and understanding the structure of behavioral dynamics. This method is particularly focused on uncovering potential hierarchical organizations within the behaviors exhibited by the system.

The premise of this study is that behaviors are naturally structured hierarchically. Therefore, increasing the number of clusters used to categorize behaviors would lead to the subdivision of existing clusters rather than a merging of behaviors from different clusters. This idea is based on the fact that behaviors in the same cluster have common characteristics or belong to the same larger behavioral group.

The behaviors are mapped into groups, $S(n) \rightarrow Z$, in a manner that compresses the current state's description while preserving critical information about the state τ transitions into the future, $S(n+\tau)$. The goal is to maximize the mutual information

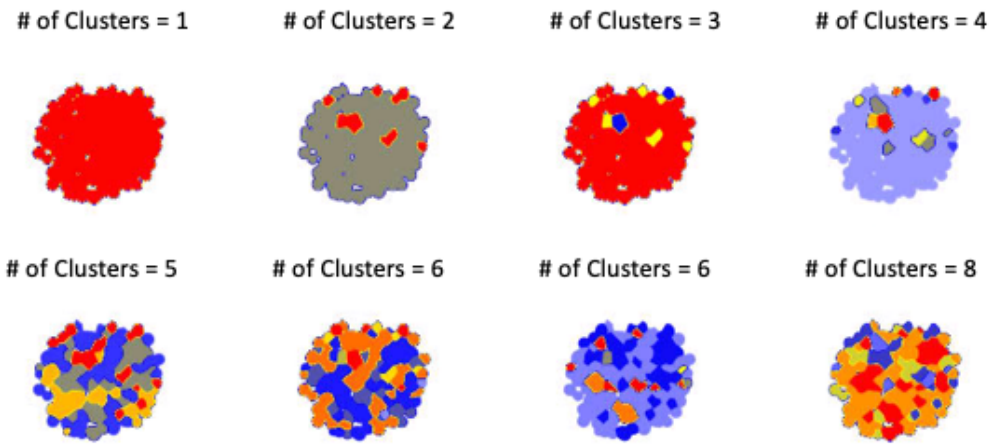


Figure 4.12: Information bottleneck partitioning of behavioral space.

between the compressed description Z and the future state $S(n+\tau)$, denoted as $I(Z; S(n+\tau))$, while maintaining a fixed level of information about the current state $S(n)$, denoted as $I(Z; S(n))$. Thus, the optimization problem can be formalized as:

$$F = I(Z; S(n+\tau)) - \beta I(Z; S(n)), \tag{4.6}$$

where β serves as a Lagrange multiplier, regulating the trade-off between the amount of information about the current state that is preserved and the predictive power regarding future states. Varying β and the number of clusters allows for an exploration of how the system's complexity can be effectively reduced without significantly compromising the ability to make accurate predictions about future states.

4.4 Discussion

In this study, we employed a sophisticated pipeline that integrates wavelet transforms, autoencoders, LSTM networks, and dimensionality reduction techniques such as t-SNE. This analysis offers a novel perspective on the stability and transitions of behavioral states. This pipeline provides insights into the stability and fluidity of

behavior. The identification of basins of attraction and the transitions between them offer a window into the mechanisms that govern behavioral changes, shedding light on how certain states become predominant and under what conditions transitions occur.

The application of wavelet transforms to behavioral data provides a rich spectral representation that captures both the frequency and temporal dynamics of behavior. This step is crucial for identifying patterns and transitions that are not immediately apparent in the raw data. By reducing the data to a more manageable form while preserving its essential features, autoencoders facilitate a deeper exploration of the behavioral states encoded within the data. The use of LSTM networks is an important decision, as they have the capability to model temporal sequences in order to predict future behavioral states. This predictive modeling serves a dual purpose: it tests the LSTM's capacity to capture the temporal dynamics of behavior and aids in the identification of fixed points within the system. The predictive nature of LSTM networks plays a significant role in understanding the evolution of behavioral states over time. Mapping the identified fixed points into a two-dimensional space via t-SNE allows for a visual representation of the basins of attraction within the behavioral dynamics. This visualization not only simplifies the interpretation of complex data but also reveals the underlying structure of behavioral states, highlighting areas of stability and potential pathways for state transitions.

Furthermore, the hierarchical organization revealed through the clustering of behavioral states underscores the modular nature of behavior. This finding suggests that complex behaviors may be constructed from simpler, foundational actions.

Chapter 5

Conclusion and Future Directions

5.1 Thesis Contributions

The primary contribution of this thesis revolves around the exploration and understanding of time-varying coupling effects on causal linkages between variables in interacting systems. To achieve this aim, we created a novel approach designed to investigate the dynamics within complex systems. This method focuses on understanding the variations in correlation and synchronization across different coupling values and is robust to datasets that are lengthy, noisy, non-linear, and non-stationary. This approach aims to fill a gap in current methods for analyzing causality in complex dynamic systems, which often struggle to accurately follow the changing patterns of variable interactions over time.

A key part of this exploration includes a comparative analysis of the proposed approach against established methods such as Granger Causality, Convergent Cross Mapping, and Transfer Entropy. It is important to note that our goal is not to discredit the effectiveness of these established techniques in areas where they have been proven to work effectively. Rather, our research focuses on trying to solve specific types of systems that have not been adequately dealt with by these conventional

methods.

We are confident in our approach due to the positive results and strong performance it has shown when tested with actual data. The results of our experiments highlight the ability of our method to detect causal relationships, outperforming current top techniques. This advancement marks a significant step forward in the ability to identify and measure causal networks from time series across various research fields. Our goal with this progress is not only to address a key gap in current tools for causal inference but also to broaden the possibilities for understanding and modeling complex causal relationships in dynamic systems.

The secondary objective of this thesis is to quantify and measure behavioral states in a multi-time-scale manner, using behavior from freely behaving rats as a test case. This study seeks to enhance our understanding by analyzing complex behavior patterns and the underlying mechanisms. The importance of this analysis lies in its ability to provide insights that are applicable across multiple animals.

This analysis is significant in contributing to the theoretical understanding of behavioral systems. It achieves this by identifying fixed points, mapping basins of attraction, and explaining the hierarchical organization of behaviors. Therefore, this research offers a novel framework for conceptualizing how complex behaviors emerge from simpler dynamics.

This theoretical advancement on behavioral states offers a promising foundation for future studies in neuroscience. By applying the same analytical pipeline developed for studying behavioral markers to brain data, we can extend our exploration into the realm of brain states, potentially uncovering a basin landscape that represents the stable and transient states of brain activity. This novel approach could revolutionize our understanding of how brain states correlate with behavior, providing insights into the neural underpinnings of complex actions and cognitive processes.

Furthermore, this analysis holds the potential to enhance predictive capabilities

regarding behavioral transitions and states. By understanding the conditions under which certain behaviors are likely to occur and how systems transition between states, we can develop strategies for predicting behavioral patterns based on historical states. Understanding the probabilistic nature of behavioral transitions allows for the prediction of how an animal is likely to behave in response to a given stimulus based on its current or previous state. This capability can significantly enhance the design of behavioral experiments, allowing researchers to design experiments to target or avoid certain behaviors more effectively.

5.2 Summaries of Chapters

5.2.1 Chapter 1 summary

This chapter sets the stage for the thesis by introducing the complex relationship between temporal dynamics, causal inference, and the analysis of behavioral states. This chapter begins with an introduction that demonstrates the significance of temporal interactions in the study of dynamic systems and their behaviors. It then explores the concept of causality, highlighting its importance in the analysis of such systems and the challenges inherent in inferring causal relationships. This leads to the introduction of Temporal Convolutional Networks (TCN) as a solution to these difficulties.

The chapter progresses to provide a comprehensive overview of TCN architecture, including the roles and functionalities of Convolutional Neural Networks (CNNs), Causal Convolutional Networks, Dilated Convolutions, Residual Connections, and Autoencoders in capturing and analyzing temporal data. This discussion serves as a prelude to the introduction of Temporal Autoencoders for Causal Inference (TACI), focusing on the TACI Autoencoder's design and its application in temporal causal inference.

The next area of focus is on behavioral analysis, including a conversation about the definition of behavior, the influence of ethology, and Tinbergen's Four Questions in comprehending behavioral motivations and mechanisms. This section also addresses the methodologies for measuring behavior and the concept of stereotyped behavior, providing a foundation for the thesis's exploration of behavioral state representations.

The chapter concludes with a thesis outline, offering readers a roadmap of the study's trajectory.

5.2.2 Chapter 2 summary

This chapter serves as a foundational overview of the key methodologies and background information used in this study. It begins by addressing the inherent challenges in causal inference, setting the stage for a deep dive into the importance of a new method that can address all the current problems in this field. The discussion follows by examining the critical distinction between correlation and causality, thereby addressing a common misconception in statistical analysis and empirical research.

Building on this foundation, we introduced in this chapter the methodology behind the most prestigious methods for causal discovery, such as "Granger Causality," "Transfer Entropy," and "Convergent Cross Mapping." We also explored the problems with these approaches and why we need to transition from traditional statistical methods to causal neural network approaches for causal inference. This segment presents the principles behind Neural Granger causality and the Temporal Causal Discovery Framework, signaling a significant leap toward integrating causality with neural network architectures. Further, we introduced a novel autoencoder architecture we designed, Temporal Autoencoders for Causal Inference (TACI), specifically for temporal causal inference.

The concluding sections of this chapter transition into the realm of behavioral analysis, offering detailed insights into the methodologies and concepts required for

the effective measurement and categorization of behavior. This includes an exploration of stereotyped behaviors and basins of attraction. Furthermore, these final sections set the stage for Chapter 4 by laying out the foundational information necessary for the innovative process we used for identifying fixed points (behavioral states) from time series data of behavior. This methodology employs recurrent neural networks (RNNs) to analyze and interpret the dynamic patterns and sequences inherent in behavioral data, enabling the identification of stable states or attractors within the system.

5.2.3 Chapter 3 summary

In this chapter, we introduced a novel approach named Temporal Autoencoders for Causal Inference (TACI), designed to address the dynamic nature of causal relationships in complex systems. Traditional causal inference methods struggle with the non-linear, non-stationary, high synchronization, and fluctuating behavior of real-world variables, where causal linkages may strengthen, weaken, or change direction over time. TACI utilizes a two-headed Temporal Convolutional Network (TCN) autoencoder to capture and analyze these causal interactions.

The foundation of TACI, known as TCN, is a type of neural network that extends the benefits of convolutional neural networks (CNNs) to time-series data. They are notable for their simplicity, ability to retain long-term memory, and effectiveness in auto-regressive prediction tasks. The neural network architecture of TACI encodes time series data of two variables, $x(t)$ and $y(t)$, and decodes a future trajectory of $y(t)$ from a compressed latent space derived from the initial time series. This method enables a detailed analysis of causal dynamics over time.

TACI uses four different network versions to determine causal inference, taking into account both the original and surrogate data to predict future states and calculate the variance explained across a moving window. We validated TACI's effectiveness

through its application to deterministic and stochastic dynamical system models with known causal relationships, where it demonstrated strong performance.

First, we applied TACI to a series of complex static artificial models by systematically varying the coupling strength between them. The systems involved were "Autoregressive Models," "Henon Maps," "Bidirectional Two Species Model," and "Rössler-Lorenz Attractors." Second, we made use of "Coupled Henon Maps" to simulate different scenarios with dynamic causal relationships. These examples included intermittent coupling, a midway flip of causal influence, and pulses of varying widths. These experiments tested TACI's ability to detect, adapt to, and accurately represent changes in causal direction and intensity over time. Lastly, we tested TACI's real-world applicability by using two datasets: the Jena Climate Dataset and electrophysiological data from a monkey undergoing anesthesia and sleep tasks. TACI successfully identified temporal seasonal coupling in the climate data and dynamic changes in causal relationships between brain areas during anesthesia induction and recovery in the monkey.

The successful application across these varied and complex scenarios demonstrates TACI's robustness and versatility in modeling time-evolving causal dynamics.

5.2.4 Chapter 4 summary

In this chapter, we conducted a thorough investigation using a detailed analytical approach to understand the intricate dynamics of behavioral states. Our investigation was not merely about identifying distinct behavioral states but aimed at understanding the transitions, stability, and hierarchical organization of these states within the broader context of the behavior's dynamics.

Initially, our attention was directed towards data preprocessing, an important stage to guarantee the accuracy and quality of the data. We use data corresponding to 3D movement kinematics encompassing the entire behavioral repertoire of rats. The

dataset utilized in this study is derived from 19 strategically placed markers on the animals, specifically targeting the head, trunk, forelimbs, and hindlimbs. We removed noise and artifacts by meticulously filtering out irrelevant or erroneous information and normalizing the data to a common scale.

We then transitioned to calculating the spectral representation of the behavioral data using wavelet transforms. This approach allowed us to capture both the frequency and temporal information contained within the data, providing a rich, multidimensional perspective on behavioral dynamics. The use of wavelets was instrumental in revealing subtle patterns and transitions not readily apparent in the raw data.

We employed an autoencoder for dimensionality reduction to manage the complexity of the spectral data. This step compressed the high-dimensional data into a more tractable form, retaining the most salient features essential for understanding the behavioral states. With the data in a reduced and concentrated form, we trained a Long Short-Term Memory (LSTM) network to predict future behavioral states. This predictive modeling served as a means to identify fixed points within the system.

Lastly, by using t-SNE, we mapped the identified fixed points into a two-dimensional space to visualize the basins of attraction. This mapping created a "basin landscape," offering a graphical representation of the stable states and the transitions between them.

5.3 Limitations

5.3.1 TACI and causal inference

The Temporal Autoencoders for Causal Inference (TACI) framework, while innovative in its approach to modeling time-varying causality in complex systems, is not without its limitations. One of the primary concerns is the extensive training time

and the resource-intensive nature of the model. Implementing TACI, especially on large datasets, requires significant computational power and time.

Another concern is the potential for overfitting due to TACI's considerable modeling capacity. While the framework is designed to capture the nuanced dynamics of causal relationships over time, like most other causal network models, this method can fit data too closely without careful regularization, resulting in models that perform exceptionally well on training data but generalize poorly to unseen data.

Furthermore, TACI incorporates elements of the Granger causality approach, which means it also inherits some of its problems. Granger causality assumes that the causal variable contains unique information about the future values of the effect variable, which might not always hold true in complex systems where numerous latent factors influence outcomes.

5.3.2 Fixed Points Pipeline

While the analytical approach detailed in this study offers significant insights into the dynamics of behavioral states, it is important to acknowledge the limitations encountered during the research process. These limitations arise from various stages of the analysis, from data collection to computational challenges, each impacting the overall effectiveness of the study.

A critical limitation arose during the data collection phase. The segments chosen for analysis were oversampled, resulting in a dataset that lacked sufficient behavioral information for an effective and comprehensive analysis. This oversampling resulted in a reduction in the quality of the behavioral data, limiting our capacity to fully capture the range of behavioral patterns. As a result, the hierarchical approach we used, which depended on the information bottleneck technique for identifying a progression from complex to simpler behaviors, did not yield the expected clarity in the hierarchical organization of behaviors. The absence of detailed behavioral

data also impeded our ability to uncover interesting findings when reviewing the flux matrix, a tool meant to illustrate the transitions between various behavioral states.

Another significant limitation was the computational demand of calculating wavelet transforms for such extensive data. Wavelet analysis, while powerful for capturing the spectral and temporal characteristics of behavioral data, requires substantial memory resources to handle effectively. The sheer volume of data, combined with the high-resolution requirements of wavelet analysis, placed considerable strain on computational resources, limiting the scope and depth of our spectral analysis.

The calculation of fixed points, a central component of our analysis aimed at identifying stable states within the behavioral dynamics, presented another challenge. Despite implementing parallel processing techniques to expedite the calculations, the process remained extremely slow. This sluggishness is inherent to the nature of fixed point calculations, which require iterative processes to allow the network's internal states to settle and converge for each data point. The time-intensive nature of reaching convergence significantly slowed down the analysis, impacting the efficiency and scalability of our approach.

5.4 Future Directions

5.4.1 Brain connectivity of prairie voles during social bonding

In this study, the Temporal Autoencoders for Causal Inference (TACI) framework can be employed to analyze the brain connectivity of prairie voles during social bonding, specifically focusing on the directional influence of one brain area over another during mating. Here, we aim to follow in the footsteps of Amadei *et al.* [129] and create a unique causal model of all the voles that allows us to explore the time-varying coupling between brain areas during social behaviors.

Despite the recognized role of mPFC–NAcc communication in coordinating behavior for reward acquisition, including the adoption of new behavioral strategies, there is limited understanding of its specific activation during affiliative behaviors. Therefore, first, we aim to validate whether mPFC–NAcc functional connectivity facilitates the transition of animals to express affiliative behavior towards a partner. Given that low-frequency drive from mPFC to NAcc can modify behavioral responses to environmental stimuli, we would like to explore if mPFC–NAcc connectivity intensifies during social behaviors that encourage more affiliative responses towards a partner.

To understand the impact of low-frequency connectivity on local activity in specific brain regions, our methodology was adapted to incorporate a spectral analysis approach. We integrated the Morlet wavelet transform into our analysis to enable the exploration of time-varying causality within the frequency domain. This advanced technique allows us to dissect the dataset's frequency domain, categorizing frequencies to assess causality with a focus on the power inherent in each time series. This spectral approach reveals the complex, evolving dynamics of interaction between brain regions over time, offering a deeper understanding of the neurological mechanisms of social bonding.

Additionally, this adjustment allows for the possibility of comparing the levels of connectivity between the medial prefrontal cortex (mPFC) and nucleus accumbens (NAcc) during various social behaviors. For instance, we can analyze the differences between these neural interactions during mating and those observed during other social activities like huddling and grooming.

5.4.2 Brain States

The extension of the fixed points (basin) pipeline to brain states involves analyzing neural activity data, such as that obtained from electroencephalography (EEG),

functional magnetic resonance imaging (fMRI), or other neuroimaging techniques, using a similar methodology to that employed for behavioral data. By calculating spectral representations, reducing dimensionality through autoencoders, and employing LSTM networks for predictive modeling, we can identify fixed points within the neural activity that correspond to stable brain states. Mapping the fixed points from these datasets into a two-dimensional basin landscape would allow us to visualize and explore the dynamics of brain states in an intuitive and informative manner.

Each basin represents a stable state or attractor within the brain's dynamical system, with transitions between basins reflecting changes in neural activity patterns. Our goal in studying these transitions is to uncover the underlying neural mechanisms of behavior. Thus, it reveals how specific brain states correlate with different actions or cognitive processes performed by animals or humans.

One interesting part of this exploration is the potential hierarchical organization of brain attractors. Just as behaviors can be broken down into simpler, modular actions, brain states may also exhibit a hierarchical structure, with complex patterns of neural activity emerging from the interaction of simpler, foundational brain states. By examining how different stable brain states interact and influence each other, we can uncover the neural networks and pathways that facilitate cognitive processes and behavioral responses.

Bibliography

[1] B. G. Veilleux. An analysis of the predatory interaction between paramecium and didinium. *Journal of Animal Ecology*, 48(3):787–803, 1979. ISSN 00218790, 13652656. URL `http://www.jstor.org/stable/4195`.

[2] Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019. ISSN 1664-8021. doi: 10.3389/fgene.2019.00524. URL `https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2019.00524`.

[3] Yifan Zhao, Steve A. Billings, Hua-Liang Wei, and Ptolemaios G. Sarrigiannis. A parametric method to measure time-varying linear and nonlinear causality with applications to eeg data. *IEEE Transactions on Biomedical Engineering*, 60(11):3141–3148, 2013. doi: 10.1109/TBME.2013.2269766.

[4] George Sugihara, Robert May, Hao Ye, Chih hao Hsieh, Ethan Deyle, Michael Fogarty, and Stephan Munch. Detecting causality in complex ecosystems. *Science*, 338(6106):496–500, 2012. doi: 10.1126/science.1227079. URL `https://www.science.org/doi/abs/10.1126/science.1227079`.

[5] RP Naidu. Causality assessment: A brief insight into practices in pharmaceutical industry. *Perspectives in Clinical Research*, 4(4):233–236, Oct 2013. doi: 10.4103/2229-3485.120173.

[6] Shawkat Hammoudeh, Ahdi Noomen Ajmi, and Khaled Mokni. Relationship between green bonds and financial and environmental variables: A novel time-varying causality. *Energy Economics*, 92:104941, 2020. ISSN 0140-9883. doi: https://doi.org/10.1016/j.eneco.2020.104941. URL https://www.sciencedirect.com/science/article/pii/S0140988320302814.

[7] J. Roderick McCrorie and Marcus J. Chambers. Granger causality and the sampling of economic processes. *Journal of Econometrics*, 132(2): 311–336, 2006. ISSN 0304-4076. doi: https://doi.org/10.1016/j.jeconom.2005.02.002. URL https://www.sciencedirect.com/science/article/pii/S0304407605000576.

[8] Rania Jammazi, Román Ferrer, Francisco Jareño, and Syed Jawad Hussain Shahzad. Time-varying causality between crude oil and stock markets: What can we learn from a multiscale perspective? *International Review of Economics and Finance*, 49:453–483, 2017. ISSN 1059-0560. doi: https://doi.org/10.1016/j.iref.2017.03.007. URL https://www.sciencedirect.com/science/article/pii/S1059056017301867.

[9] M Wang, Z Liao, D Mao, Q Zhang, Y Li, E Yu, and Z Ding. Application of granger causality analysis of the directed functional connection in alzheimer's disease and mild cognitive impairment. *Journal of Visualized Experiments*, Aug 2017. doi: 10.3791/56015.

[10] L. Hao, Z. Sheng, W. Ruijun, et al. Altered granger causality connectivity within motor-related regions of patients with parkinson's disease: a resting-state fmri study. *Neuroradiology*, 62:63–69, 2020. doi: 10.1007/s00234-019-02311-z. URL https://doi.org/10.1007/s00234-019-02311-z.

[11] T. Paus. Inferring causality in brain images: a perturbation approach. *Philo-*

sophical Transactions of the Royal Society of London. Series B: Biological Sciences, 360(1457):1109–1114, May 2005. doi: 10.1098/rstb.2005.1652.

[12] Mehrdad Jazayeri and Arash Afraz. Navigating the neural space in search of the neural code. *Neuron*, 93(5):1003–1014, 2017. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2017.02.019. URL https://www.sciencedirect.com/science/article/pii/S0896627317301034.

[13] Steffen BE Wolff and Bence P Ölveczky. The promise and perils of causal circuit manipulations. *Current Opinion in Neurobiology*, 49:84–94, 2018. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2018.01.004. URL https://www.sciencedirect.com/science/article/pii/S0959438817302246. Neurobiology of Behavior.

[14] Jakob Runge, Sebastian Bathiany, Erik Bollt, et al. Inferring causation from time series in earth system sciences. *Nature Communications*, 10:2553, 2019. doi: 10.1038/s41467-019-10105-3. URL https://doi.org/10.1038/s41467-019-10105-3.

[15] van Nes, H. Egbert, Marten Scheffer, Victor Brovkin, Timothy M. Lenton, Hao Ye, Ethan Deyle, and George Sugihara. Causal feedbacks in climate change. *Nature Climate Change*, 5(5):445–448, May 2015. ISSN 1758-6798. URL https://doi.org/10.1038/nclimate2568.

[16] Jakob Runge, Peer Nowack, Marlene Kretschmer, Seth Flaxman, and Dino Sejdinovic. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances*, 5(11):eaau4996, 2019. doi: 10.1126/sciadv.aau4996. URL https://www.science.org/doi/abs/10.1126/sciadv.aau4996.

[17] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition, 2009. ISBN 978-0-521-89560-6.

[18] Hernando Ombao and SÉbastien Van Bellegem. Evolutionary coherence of nonstationary signals. *IEEE Transactions on Signal Processing*, 56(6):2259–2266, 2008. doi: 10.1109/TSP.2007.914341.

[19] Ethan R. Deyle and George Sugihara. Generalized theorems for nonlinear state space reconstruction. *PLOS ONE*, 6(3):1–8, 03 2011. doi: 10.1371/journal.pone.0018295. URL `https://doi.org/10.1371/journal.pone.0018295`.

[20] Yifan Zhao, Steve A. Billings, Hualiang Wei, and Ptolemaios G. Sarrigiannis. Tracking time-varying causality and directionality of information flow using an error reduction ratio test with applications to electroencephalography data. *Phys. Rev. E*, 86:051919, Nov 2012. doi: 10.1103/PhysRevE.86.051919. URL `https://link.aps.org/doi/10.1103/PhysRevE.86.051919`.

[21] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969. ISSN 00129682, 14680262. URL `http://www.jstor.org/stable/1912791`.

[22] Thomas Schreiber. Measuring information transfer. *Phys. Rev. Lett.*, 85:461–464, Jul 2000. doi: 10.1103/PhysRevLett.85.461. URL `https://link.aps.org/doi/10.1103/PhysRevLett.85.461`.

[23] Sarah Cobey and Edward B. Baskerville. Limits to causal inference with state-space reconstruction for infectious disease. *PLOS ONE*, 11(12):1–22, 12 2016. doi: 10.1371/journal.pone.0169050. URL `https://doi.org/10.1371/journal.pone.0169050`.

[24] Yonghong Chen, Govindan Rangarajan, Jianfeng Feng, and Mingzhou Ding. Analyzing multiple nonlinear time series with extended Granger causality.

Physics Letters A, 324(1):26–35, April 2004. ISSN 03759601. doi: 10.1016/j.physleta.2004.02.032. URL https://linkinghub.elsevier.com/retrieve/pii/S0375960104002403.

[25] D. T. Tyler Flockhart, Jean-Baptiste Pichancourt, D. Ryan Norris, and Tara G. Martin. Unravelling the annual cycle in a migratory animal: breeding-season habitat loss drives population declines of monarch butterflies. *Journal of Animal Ecology*, 84(1):155–165, 2015. doi: https://doi.org/10.1111/1365-2656.12253. URL https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/1365-2656.12253.

[26] Christelle Lasbleiz, Jean-François Ferveur, and Claude Everaerts. Courtship behaviour of drosophila melanogaster revisited. *Animal Behaviour*, 72(5):1001–1012, 2006. ISSN 0003-3472. doi: https://doi.org/10.1016/j.anbehav.2006.01.027. URL https://www.sciencedirect.com/science/article/pii/S0003347206002703.

[27] Gordon J. Berman, Daniel M. Choi, William Bialek, and Joshua W. Shaevitz. Mapping the stereotyped behaviour of freely moving fruit flies. *Journal of The Royal Society Interface*, 11:20140672, 2014. doi: 10.1098/rsif.2014.0672. URL http://doi.org/10.1098/rsif.2014.0672.

[28] GJ Berman, W Bialek, and JW Shaevitz. Predictability and hierarchy in drosophila behavior. *Proceedings of the National Academy of Sciences*, 113(42):11943–11948, 2016.

[29] A. B. Wiltschko, M. J. Johnson, G. Iurilli, R. E. Peterson, J. M. Katon, S. L. Pashkovski, V. E. Abraira, R. P. Adams, and S. R. Datta. Mapping sub-second structure in mouse behavior. *Neuron*, 88(6):1121–1135, Dec 2015. doi: 10.1016/j.neuron.2015.11.031.

[30] Greg J. Stephens, Bethany Johnson-Kerner, William Bialek, and William S. Ryu. Dimensionality and dynamics in the behavior of c. elegans. *PLOS Computational Biology*, 4(4):1–10, 04 2008. doi: 10.1371/journal.pcbi.1000028. URL https://doi.org/10.1371/journal.pcbi.1000028.

[31] Gordon J Berman. Measuring behavior across scales. *BMC Biology*, 16 (1):23, 2018. doi: 10.1186/s12915-018-0494-7. URL https://bmcbiol.biomedcentral.com/articles/10.1186/s12915-018-0494-7.

[32] David Sussillo and Omri Barak. Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Computation*, 25(3):626–649, 2013. doi: 10.1162/NECO_a_00409.

[33] Zhengxiong Wang and Anton Ragni. Approximate fixed-points in recurrent neural networks, 2021.

[34] Atle Mysterud, Nils Chr Stenseth, Nigel G Yoccoz, Rolf Langvatn, and Geir Steinheim. Nonlinear effects of large-scale climatic variability on wild and domestic herbivores. *Nature*, 410(6832):1096–1099, 2001.

[35] Chih-hao Hsieh, Sarah M. Glaser, Andrew J. Lucas, and George Sugihara. Distinguishing random environmental fluctuations from ecological catastrophes for the north pacific ocean. *Nature*, 435(7040):336–340, May 2005. ISSN 1476-4687. URL https://doi.org/10.1038/nature03553.

[36] Robert M. May, Simon A. Levin, and George Sugihara. Ecology for bankers. *Nature*, 451(7181):893–894, February 2008. ISSN 1476-4687. URL https://doi.org/10.1038/451893a.

[37] Furkan Emirmahmutoglu, Zulal Denaux, and Mert Topcu. Time-varying causality between renewable and non-renewable energy consumption and real output:

Sectoral evidence from the united states. *Renewable and Sustainable Energy Reviews*, 149:111326, 2021. ISSN 1364-0321. doi: https://doi.org/10.1016/j.rser.2021.111326. URL https://www.sciencedirect.com/science/article/pii/S1364032121006122.

[38] International Council of Scientific Unions. Scientific Committee on Oceanic Research and International Council of Scientific Unions. Scientific Committee on Oceanic Research. Canadian National Committee. *Proceedings of the Joint Oceanographic Assembly 1982 General Symposia: Dalhousie University, Halifax, Nova Scotia, Canada*. Canadian National Committee for the Scientific Committee on Oceanic Research . . . , 1983.

[39] R. Quian Quiroga, J. Arnhold, and P. Grassberger. Learning driver-response relationships from synchronization patterns. *Phys. Rev. E*, 61:5142–5148, May 2000. doi: 10.1103/PhysRevE.61.5142. URL https://link.aps.org/doi/10.1103/PhysRevE.61.5142.

[40] Nikola Jajcay, Sergey Kravtsov, George Sugihara, Anastasios A. Tsonis, and Milan Paluš. Synchronization and causality across time scales in el niño southern oscillation. *npj Climate and Atmospheric Science*, 1(1):33, Nov 2018. ISSN 2397-3722. doi: 10.1038/s41612-018-0043-7. URL https://doi.org/10.1038/s41612-018-0043-7.

[41] S. Boccaletti, J. Kurths, G. Osipov, D. L. Valladares, and C. S. Zhou. The synchronization of chaotic systems. *Physics Reports*, 366(1–2):1–101, 2002. ISSN 0370-1573. doi: 10.1016/S0370-1573(02)00137-0. URL https://www.sciencedirect.com/science/article/pii/S0370157302001370.

[42] Daniel A Levitis, William Z Lidicker, and Glenn Freund. Behavioural biologists don't agree on what constitutes behaviour. *Animal Behaviour*, 78

(1):103–110, 2009. doi: 10.1016/j.anbehav.2009.03.018. URL https://www.sciencedirect.com/science/article/pii/S0003347209001451.

[43] N. Tinbergen. *The Study of Instinct*. Clarendon Press/Oxford University Press, 1951.

[44] N. Tinbergen. On aims and methods of ethology. *Zeitschrift für Tierpsychologie*, 20(4):410–433, 1963. doi: https://doi.org/10.1111/j.1439-0310.1963.tb01161.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1439-0310.1963.tb01161.x.

[45] S.E. Roian Egnor and Kristin Branson. Computational analysis of behavior. *Annual Review of Neuroscience*, 39(Volume 39, 2016): 217–236, 2016. ISSN 1545-4126. doi: https://doi.org/10.1146/annurev-neuro-070815-013845. URL https://www.annualreviews.org/content/journals/10.1146/annurev-neuro-070815-013845.

[46] Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current Opinion in Neurobiology*, 32:148–155, 2015. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2015.04.003. URL https://www.sciencedirect.com/science/article/pii/S0959438815000768. Large-Scale Recording Technology (32).

[47] R. B. Nelsen. Kendall tau metric. *Encyclopedia of Mathematics*, 2001. URL http://encyclopediaofmath.org/index.php?title=Kendall_tau_metric&oldid=52761. Originally published 1994.

[48] Naomi Altman and M. Krzywinski. Association, correlation and causation. *Nature Methods*, 12(10):899–900, 2015. doi: 10.1038/nmeth.3587.

[49] Giovanni Comandé. Opinions. the rotting meat error: From galileo to aristotle

in data mining. *European Data Protection Law Review*, 4(3), 2018. doi: 10.21552/edpl/2018/3/4. URL https://doi.org/10.21552/edpl/2018/3/4.

[50] C. W. J. Granger. Time series analysis, cointegration, and applications. Nobel Lecture, December 8, 2003, in: *Les Prix Nobel. The Nobel Prizes 2003*, 2004. URL http://nobelprize.org/nobel_prizes/economics/laureates/2003/granger-lecture.pdf.

[51] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 37(3):424–438, 1969. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1912791.

[52] L Schiatti, G Nollo, G Rossato, and L Faes. Extended granger causality: a new tool to identify the structure of physiological networks. *Physiological Measurement*, 36(4):827, mar 2015. doi: 10.1088/0967-3334/36/4/827. URL https://dx.doi.org/10.1088/0967-3334/36/4/827.

[53] Yonghong Chen, Govindan Rangarajan, Jianfeng Feng, and Mingzhou Ding. Analyzing multiple nonlinear time series with extended granger causality. *Physics Letters A*, 324(1):26–35, April 2004. ISSN 0375-9601. doi: 10.1016/j.physleta.2004.02.032. URL http://dx.doi.org/10.1016/j.physleta.2004.02.032.

[54] Floris Takens. Detecting strange attractors in turbulence. In David Rand and Lai-Sang Young, editors, *Dynamical Systems and Turbulence, Warwick 1980*, pages 366–381, Berlin, Heidelberg, 1981. Springer Berlin Heidelberg. ISBN 978-3-540-38945-3.

[55] J. P. Eckmann and D. Ruelle. Ergodic theory of chaos and strange attractors. *Rev. Mod. Phys.*, 57:617–656, Jul 1985. doi: 10.1103/RevModPhys.57.617. URL https://link.aps.org/doi/10.1103/RevModPhys.57.617.

[56] Anna Krakovská, Jozef Jakubík, Martina Chvosteková, David Coufal, Nikola Jajcay, and Milan Paluš. Comparison of six methods for the detection of causality in a bivariate time series. *Phys. Rev. E*, 97:042207, Apr 2018. doi: 10.1103/PhysRevE.97.042207. URL `https://link.aps.org/doi/10.1103/PhysRevE.97.042207`.

[57] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x.

[58] N. Wiener. The theory of prediction. In E. F. Beckenbach, editor, *Modern Mathematics for Engineers*. McGraw-Hill, New York, 1956.

[59] R. Vicente, M. Wibral, M. Lindner, et al. Transfer entropy—a model-free measure of effective connectivity for the neurosciences. *Journal of Computational Neuroscience*, 30:45–67, 2011. doi: 10.1007/s10827-010-0262-3. URL `https://doi.org./10.1007/s10827-010-0262-3`.

[60] Katerina Hlaváčková-Schindler, Milan Paluš, Martin Vejmelka, and Joydeep Bhattacharya. Causality detection based on information-theoretic approaches in time series analysis. *Physics Reports*, 441(1):1–46, 2007. ISSN 0370-1573. doi: https://doi.org/10.1016/j.physrep.2006.12.004. URL `https://www.sciencedirect.com/science/article/pii/S0370157307000403`.

[61] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951. doi: 10.1214/aoms/1177729694. URL `https://doi.org/10.1214/aoms/1177729694`.

[62] Milan Paluš, Vladimír Komárek, Zbyn ěk Hrnčíř, and Katalin Štěrbová. Synchronization as adjustment of information rates: detection from bivariate time

series. *Phys. Rev. E*, 63:046211, Mar 2001. doi: 10.1103/PhysRevE.63.046211. URL https://link.aps.org/doi/10.1103/PhysRevE.63.046211.

[63] Joseph Lizier. Java information dynamics toolkit (jidt), 2023. URL https://github.com/jlizier/jidt. GitHub repository.

[64] Paul A. Dixon, Maria J. Milicich, and George Sugihara. Episodic fluctuations in larval supply. *Science*, 283(5407):1528–1530, 1999. doi: 10.1126/science.283.5407.1528. URL https://www.science.org/doi/abs/10.1126/science.283.5407.1528.

[65] Floris Takens. Detecting strange attractors in turbulence. In David Rand and Lai-Sang Young, editors, *Dynamical Systems and Turbulence, Warwick 1980*, pages 366–381, Berlin, Heidelberg, 1981. Springer Berlin Heidelberg. ISBN 978-3-540-38945-3.

[66] J.P. Crutchfield. Observing complexity and the complexity of observation. In H. Atmanspacher and G.J. Dalenoort, editors, *Inside Versus Outside*, volume 63 of *Springer Series in Synergetics*. Springer, Berlin, Heidelberg, 1994. doi: 10.1007/978-3-642-48647-0_14. URL https://doi.org/10.1007/978-3-642-48647-0_14.

[67] T. Sauer, J.A. Yorke, and M. Casdagli. Embedology. *Journal of Statistical Physics*, 65:579–616, 1991. doi: 10.1007/BF01053745. URL https://doi.org/10.1007/BF01053745.

[68] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Phys. Rev. Lett.*, 45:712–716, Sep 1980. doi: 10.1103/PhysRevLett.45.712. URL https://link.aps.org/doi/10.1103/PhysRevLett.45.712.

[69] Kresimir Josic. Synchronization of chaotic systems and invariant manifolds.

Nonlinearity, 13(4):1321, jul 2000. doi: 10.1088/0951-7715/13/4/318. URL https://dx.doi.org/10.1088/0951-7715/13/4/318.

[70] D. Ruelle. *Chaotic Evolution and Strange Attractors*. Lezioni Lincee. Cambridge University Press, 1989. doi: 10.1017/CBO9780511608773.

[71] Chuan Luo, Xiaolong Zheng, and Daniel Zeng. Causal inference in social media using convergent cross mapping. In *2014 IEEE Joint Intelligence and Security Informatics Conference*, pages 260–263, 2014. doi: 10.1109/JISIC.2014.50.

[72] G. Sugihara and R. May. Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, 344:734–741, 1990. doi: 10.1038/344734a0. URL https://doi.org/10.1038/344734a0.

[73] Keichi Takahashi, Wassapon Watanakeesuntorn, Kohei Ichikawa, Joseph Park, Ryousei Takano, Jason Haga, George Sugihara, and Gerald M. Pao. kedm: A performance-portable implementation of empirical dynamic modeling using kokkos. In *Practice and Experience in Advanced Research Computing (PEARC '21)*, pages 1–8, Boston, MA, USA, July 2021. ACM. doi: 10.1145/3437359.3465571. URL https://doi.org/10.1145/3437359.3465571.

[74] Ragav Venkatesan and Baoxin Li. *Convolutional Neural Networks in Visual Computing: A Concise Guide*. CRC Press, 2017. ISBN 978-1-351-65032-8.

[75] R. Yamashita, M. Nishio, R.K.G. Do, et al. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9:611–629, 2018. doi: 10.1007/s13244-018-0639-9. URL https://doi.org/10.1007/s13244-018-0639-9.

[76] Alex Tank, Ian Covert, Nicholas Foti, Ali Shojaie, and Emily B. Fox. Neural granger causality. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):4267–4279, 2022. doi: 10.1109/TPAMI.2021.3065601.

[77] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Series in Statistics. Springer, New York, NY, USA, 2 edition, 2009. ISBN 978-0-387-84857-0. doi: 10.1007/978-0-387-84858-7. URL https://doi.org/10.1007/978-0-387-84858-7. eBook ISBN 978-0-387-84858-7. Published: 26 August 2009. Hardcover published: 09 February 2009.

[78] Colin Lea, René Vidal, Austin Reiter, and Gregory D. Hager. Temporal convolutional networks: A unified approach to action segmentation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 47–54, Cham, 2016. Springer International Publishing. ISBN 978-3-319-49409-8.

[79] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *CoRR*, abs/1803.01271, 2018. URL http://arxiv.org/abs/1803.01271.

[80] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016. URL http://arxiv.org/abs/1609.03499.

[81] Meike Nauta, Doina Bucur, and Christin Seifert. Causal discovery with attention-based convolutional neural networks. *Machine Learning and Knowledge Extraction*, 1(1):312–340, 2019. ISSN 2504-4990. doi: 10.3390/make1010019. URL https://www.mdpi.com/2504-4990/1/1/19.

[82] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

[83] M. Kramer. Nonlinear principal component analysis using autoassociative

neural networks. *AIChE Journal*, 37(2):233–243, 1991. doi: 10.1002/AIC.690370209.

[84] Jeremy Jordan. Autoencoders, 2017. URL `https://www.jeremyjordan.me/autoencoders/`. Accessed: 2024-03-31.

[85] G.C. Layek. *An Introduction to Dynamical Systems and Chaos*. Springer, New Delhi; Heidelberg; New York; Dordrecht; London, 2015. ISBN 978-81-322-2555-3. doi: 10.1007/978-81-322-2556-0.

[86] George Datseris and Alexandre Wagemakers. Effortless estimation of basins of attraction. *Chaos*, 32(2):023104, Feb 2022. doi: 10.1063/5.0076568. URL `https://doi-org.proxy.library.emory.edu/10.1063/5.0076568`.

[87] J. C. Sprott and Anda Xiong. Classifying and quantifying basins of attraction. *Chaos*, 25(8):083101, Aug 2015. doi: 10.1063/1.4927643. URL `https://doi.org/10.1063/1.4927643`.

[88] Franz M. Rohrhofer, Stefan Posch, Clemens Gößnitzer, and Bernhard C. Geiger. On the role of fixed points of dynamical systems in training physics-informed neural networks, 2023.

[89] John J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci U S A*, 79(8):2554–2558, Apr 1982. doi: 10.1073/pnas.79.8.2554.

[90] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[91] Christopher Olah. Understanding lstm networks. Colah's Blog, Aug 2015. URL `http://colah.github.io/posts/2015-08-Understanding-LSTMs/`.

[92] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[93] Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara. Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1):388–427, 2021. ISSN 0169-2070. doi: https://doi.org/10.1016/j.ijforecast.2020.06.008. URL https://www.sciencedirect.com/science/article/pii/S0169207020300996.

[94] P. Goupillaud, A. Grossmann, and J. Morlet. Cycle-octave and related transforms in seismic signal analysis. *Geoexploration*, 23(1):85–102, 1984. ISSN 0016-7142. doi: https://doi.org/10.1016/0016-7142(84)90025-5. URL https://www.sciencedirect.com/science/article/pii/0016714284900255. Seismic Signal Analysis and Discrimination III.

[95] Ingrid Daubechies. *Ten Lectures on Wavelets*. Society for Industrial and Applied Mathematics, 1992. doi: 10.1137/1.9781611970104. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611970104.

[96] Quentin Fournier and Daniel Aloise. Empirical comparison between autoencoders and traditional dimensionality reduction methods. In *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*. IEEE, June 2019. doi: 10.1109/aike.2019.00044. URL http://dx.doi.org/10.1109/AIKE.2019.00044.

[97] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9:85, 2008.

[98] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000. doi: 10.1126/

science.290.5500.2323. URL https://www.science.org/doi/abs/10.1126/science.290.5500.2323.

[99] Michael A. A. Cox and Trevor F. Cox. *Multidimensional Scaling*, pages 315–347. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-33037-0. doi: 10.1007/978-3-540-33037-0_14. URL https://doi.org/10.1007/978-3-540-33037-0_14.

[100] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. doi: 10.1126/science.290.5500.2319. URL https://www.science.org/doi/abs/10.1126/science.290.5500.2319.

[101] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, Hoboken, NJ, 2nd edition, 2006.

[102] György Buzsáki. *Rhythms of the Brain*. Oxford University Press, New York, online edn, oxford academic, 1 may 2009 edition, 2006. doi: 10.1093/acprof:oso/9780195301069.001.0001.

[103] M. Latif, R. Kleeman, and C. Eckert. Greenhouse warming, decadal variability, or el niño? an attempt to understand the anomalous 1990s. *Journal of Climate*, 10(9):2221 – 2239, 1997. doi: 10.1175/1520-0442(1997)010⟨2221:GWDVOE⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/clim/10/9/1520-0442_1997_010_2221_gwdvoe_2.0.co_2.xml.

[104] Xin Wang, Dongxiao Wang, and Wen Zhou. Decadal variability of twentieth-century el niño and la niña occurrence from observations and ipcc ar4 coupled models. *Geophysical Research Letters*, 36(11), 2009. doi: https://doi.org/10.1029/2009GL037929. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2009GL037929.

[105] Patrick A. Stokes and Patrick L. Purdon. A study of problems encountered in granger causality analysis from a neuroscience perspective. *Proceedings of the National Academy of Sciences*, 114(34):E7063–E7072, 2017. doi: 10.1073/pnas.1704663114. URL https://www.pnas.org/doi/abs/10.1073/pnas.1704663114.

[106] Alex Eric Yuan and Wenying Shou. Data-driven causal analysis of observational biological time series. *eLife*, 11:e72518, aug 2022. ISSN 2050-084X. doi: 10.7554/eLife.72518. URL https://doi.org/10.7554/eLife.72518.

[107] Dan Mønster, Riccardo Fusaroli, Kristian Tylén, Andreas Roepstorff, and Jacob F. Sherson. Causal inference from noisy time-series data — testing the convergent cross-mapping algorithm in the presence of noise and external influence. *Future Generation Computer Systems*, 73:52–62, 2017. ISSN 0167-739X. doi: 10.1016/j.future.2016.12.009. URL https://www.sciencedirect.com/science/article/pii/S0167739X16307427.

[108] Hao Ye, Ethan R. Deyle, Luis J. Gilarranz, and George Sugihara. Distinguishing time-delayed causal interactions using convergent cross mapping. *Scientific Reports*, 5(1):14750, 2015. doi: 10.1038/srep14750. URL https://doi.org/10.1038/srep14750.

[109] Daniele Marinazzo, Mario Pellicoro, and Sebastiano Stramaglia. Kernel method for nonlinear granger causality. *Phys. Rev. Lett.*, 100:144103, Apr 2008. doi: 10.1103/PhysRevLett.100.144103. URL https://link.aps.org/doi/10.1103/PhysRevLett.100.144103.

[110] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. Causal effect inference with deep latent-variable models. *Advances in neural information processing systems*, 30, 2017.

[111] Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michèle Sebag. Causal generative neural networks. *arXiv preprint arXiv:1711.08936*, 2017.

[112] NickC1. skccm: State-space reconstruction by k-nearest neighbors convergent cross mapping. `https://github.com/nickc1/skccm/blob/master/skccm/skccm.py`, 2017.

[113] Joseph T. Lizier. Jidt: An information-theoretic toolkit for studying the dynamics of complex systems. *Frontiers in Robotics and AI*, 1:11, 2014. doi: 10.3389/frobt.2014.00011. Pre-print: arXiv:1408.3270.

[114] Edward Norton Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963. doi: 10.1175/1520-0469(1963)020⟨0130:DNF⟩2.0.CO;2.

[115] O. E. Rössler. An equation for continuous chaos. *Physics Letters A*, 57(5):397–398, 1976. doi: 10.1016/0375-9601(76)90101-8.

[116] Michel Le Van Quyen, Jacques Martinerie, Claude Adam, and Francisco J. Varela. Nonlinear analyses of interictal eeg map the brain interdependences in human focal epilepsy. *Physica D: Nonlinear Phenomena*, 127(3):250–266, 1999. ISSN 0167-2789. doi: https://doi.org/10.1016/S0167-2789(98)00258-9. URL `https://www.sciencedirect.com/science/article/pii/S0167278998002589`.

[117] Michel Hénon. A two-dimensional mapping with a strange attractor. *Communications in Mathematical Physics*, 50(1):69–77, 1976. doi: 10.1007/BF01608556.

[118] Milan Paluš and Martin Vejmelka. Directionality of coupling from bivariate time series: How to avoid false causalities and missed connections. *Phys. Rev.*

E, 75:056211, May 2007. doi: 10.1103/PhysRevE.75.056211. URL https://link.aps.org/doi/10.1103/PhysRevE.75.056211.

[119] Steven J. Schiff, Paul So, Taeun Chang, Robert E. Burke, and Tim Sauer. Detecting dynamical interdependence and generalized synchrony through mutual prediction in a neural ensemble. *Phys. Rev. E*, 54:6708–6724, Dec 1996. doi: 10.1103/PhysRevE.54.6708. URL https://link.aps.org/doi/10.1103/PhysRevE.54.6708.

[120] Baligh Mnassri. Jena climate dataset. https://www.bgc-jena.mpg.de/wetter/, 2019.

[121] M. G. Lawrence. The relationship between relative humidity and the dewpoint temperature in moist air: A simple conversion and applications. *Bulletin of the American Meteorological Society*, 86(2):225–234, 2005. doi: 10.1175/BAMS-86-2-225. URL https://doi.org/10.1175/BAMS-86-2-225.

[122] Toru Yanagawa, Zenas C. Chao, Naomi Hasegawa, and Naotaka Fujii. Large-scale information flow in conscious and unconscious states: an ecog study in monkeys. *PLOS ONE*, 8(11):null, 11 2013. doi: 10.1371/journal.pone.0080845. URL https://doi.org/10.1371/journal.pone.0080845.

[123] Satoshi Tajima, Toru Yanagawa, Naotaka Fujii, and Taro Toyoizumi. Untangling brain-wide dynamics in consciousness by cross-embedding. *PLoS Comput Biol*, 11(11):e1004537, 2015. doi: 10.1371/journal.pcbi.1004537. URL https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1004537.

[124] Yasuo Nagasaka, Kentaro Shimoda, and Naotaka Fujii. Multidimensional recording (mdr) and data sharing: An ecological open research and educational

platform for neuroscience. *PLOS ONE*, 6(7):1–7, 07 2011. doi: 10.1371/journal.pone.0022561. URL https://doi.org/10.1371/journal.pone.0022561.

[125] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[126] C. Weinreb, J. Pearl, S. Lin, MAM Osman, L. Zhang, S. Annapragada, E. Conlin, R. Hoffman, S. Makowska, WF Gillis, M. Jay, S. Ye, A. Mathis, MW Mathis, T. Pereira, SW Linderman, and SR Datta. Keypoint-moseq: parsing behavior by linking point tracking to pose dynamics. *bioRxiv*, page 2023.03.16.532307, Dec 2023. doi: 10.1101/2023.03.16.532307.

[127] Stefan Banach. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3:133–181, 1922. doi: 10.4064/fm-3-1-133-181. URL https://web.archive.org/web/20110607000000/http://matwbn.icm.edu.pl/ksiazki/fm/fm3/fm3120.pdf. Archived (PDF) from the original on 2011-06-07.

[128] Jesse D. Marshall, Diego E. Aldarondo, Timothy W. Dunn, William L. Wang, Gordon J. Berman, and Bence P. Ölveczky. Continuous whole-body 3d kinematic recordings across the rodent behavioral repertoire. *Neuron*, 109(3): 420–437.e8, 2021. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2020.11.016. URL https://www.sciencedirect.com/science/article/pii/S0896627320308941.

[129] Elizabeth A Amadei, Zachary V Johnson, Yong Jun Kwon, Aaron C Shpiner, Varun Saravanan, Wittney D Mays, Steven J Ryan, Hasse Walum, Donald G Rainnie, Larry J Young, et al. Dynamic corticostriatal activity biases social bonding in monogamous female prairie voles. *Nature*, 546(7657):297–301, 2017.