UNIVERSITAT DE BARCELONA

Final Degree Project

Biomedical Engineering Degree

Benchmarking Latent Variable Models for Neural Representations

Benchmarking Models de Variable Latent per Representacions Neurals

Barcelona, 09 de 06 de 2026

Author: David García Ortiz

Director/s: Jordi Abante Llenas

Tutor: Jordi Abante Llenas

Abstract

Computational neuroscience seeks to understand how internal brain mechanisms encode behavior and sensory stimuli through mathematical models that capture the dynamics of neural populations. Latent variable models are particularly useful because they identify low-dimensional representations from recordings of thousands of neurons, enabling visualization of how coordinated neural activity relates to behavior or external stimuli. This thesis benchmarks deep learning-based generalizations (CEBRA, GP-VAE) with baseline models (PCA, UMAP, t-SNE) on two neural recording datasets with positional context labels, involving navigation and proprioception tasks. The evaluated models included a generative approach consisting of a modified variational autoencoder incorporating a Gaussian process prior and a contrastive term, as well as non-generative CEBRA models in both supervised and self-supervised modes. Model performance was assessed through the decoding performance of the produced embeddings from the models on predicting the positional label and a global activity metric quantifying the amount of information retained from the original neural signal. Models were compared using a Pareto analysis that jointly optimized the coefficient of determination for both objectives while accounting for latent dimensionality to evaluate representational compactness.

Overall, the results indicate that PCA with a sufficient number of dimensions provides the best balance between information preservation and decoding performance on these datasets. Although deep latent variable models achieved superior decoding performance, particularly in supervised settings, they were penalized by their lower ability to preserve the global structure of the original signal.

Keywords: Neural Manifold, Latent Variable Models, Decoding Performance, Contrastive Learning, Pareto Analysis

Resum

La neurociència computacional busca entendre com els mecanismes interns del cervell codifiquen el comportament i els estímuls sensorials mitjançant models matemàtics que capturen la dinàmica de les poblacions neuronals. Els models de variables latents són especialment útils perquè identifiquen representacions de baixa dimensionalitat a partir d'enregistraments de milers de neurones, cosa que permet visualitzar com l'activitat neuronal coordinada es relaciona amb el comportament o amb estímuls externs. Aquesta tesi compara generalitzacions basades en aprenentatge profund (CEBRA, GP-VAE) amb models de referència (PCA, UMAP i t-SNE) en dos conjunts de dades d'enregistraments neuronals amb etiquetes de context posicional, corresponents a tasques de navegació i propiocepció. Els models avaluats inclouen un enfocament generatiu basat en un autoencoder variacional modificat que incorpora un prior de procés gaussià i un terme contrastiu, així com models CEBRA no generatius tant en mode supervisat com autosupervisat. El rendiment dels models es va avaluar mitjançant la capacitat de descodificació de les representacions latents generades per predir l'etiqueta posicional, així com a través d'una mètrica global d'activitat que quantifica la quantitat d'informació conservada del senyal neuronal original. Els models es van comparar utilitzant una anàlisi de Pareto que optimitzava conjuntament el coeficient de determinació per a tots dos objectius, tenint en compte també la dimensionalitat latent per avaluar la compacitat de les representacions.

En conjunt, els resultats indiquen que la PCA, amb un nombre suficient de dimensions, ofereix el millor equilibri entre la preservació de la informació i el rendiment de descodificació en aquests conjunts de dades. Tot i que els models profunds de variables latents van assolir un millor rendiment de descodificació, especialment en entorns supervisats, van ser penalitzats per la seva menor capacitat de preservar l'estructura global del senyal original.

Paraules Clau: Manifold Neural, Models de Variable Latent, Rendiment de Descodificació, Contrastive Learning, Anàlisi de Pareto

Acknowledgements

I would like to express my sincere gratitude to Jordi Abante and Berta Ros for their guidance and supervision throughout this project. Their continuous feedback, support and encouragement have been especially valuable during this work. They motivated me to further pursue this field of research and helped me to understand those concepts that were more difficult to grasp.

Beyond their academic guidance, they helped me gain confidence in my abilities and encouraged me to pursue my future goals with ambition and determination. I am especially grateful for their patience, optimism and support during the more challenging moments of the project. Their support allowed me to face difficulties with greater calm and perspective.

It has been a privilege to learn from and work alongside them, and I sincerely hope to have the opportunity to continue collaborating with them in the future.

List of Figures

1 The evolution of population dynamics from neural recordings are constrained within a manifold embedded in a latent space. Schematic by Sadtler et al. [37] . . . 11
2 Biophysical properties and processes constraints the population states into a neural manifolds. Schematic by Perich et al. [16] . . . 13
3 Variational autoencoders encoder-decoder structure. Figure from Odaibo et al. [50] . . . 19
4 Contrastive learning data sampling scheme of CEBRA model. Adapted from [9]. . . . 22
5 Sampling scheme of CEBRA for time only and position label models. Adapted from [9]. . . . 24
6 Example of neural activity and behavioral labels at the first 1000 timesteps extracted from the Achilles rat hippocampus dataset. Adapted from Grosmark et al. [65]. . . . 29
7 Hand position trajectories during experimental session of the monkey. Adapted from Chowdhury et al. [67]. . . . 30
8 Barplot of the best R^2 scores for position decoding of each model variety using non-linear decoders over latent dimensions for rat hippocampus dataset . . . 44
9 Barplot of the best R^2 scores for global activity reconstruction of each model variety using XGBoost estimators over latent dimensions for rat hippocampus dataset. . . . 44
10 Pareto Front Analysis for non-linear R^2 scores for global signal reconstruction and position decoding in rat hippocampus dataset. . . . 49
11 Barplot of the best R^2 scores for position decoding of each model variety using XGBoost estimators over latent dimensions for macaque reaching dataset. . . . 52
12 Barplot of the best R^2 scores for global activity reconstruction of each model variety using XGBoost estimators over latent dimensions for macaque reaching dataset. . . . 53
13 Pareto Front Analysis for non-linear R^2 scores for global signal reconstruction and position decoding in macaque reaching dataset. . . . 56
14 WBS diagram illustrating the technical phases of the project . . . 60
15 GANTT chart illustrating the timeline progression of the project . . . 61
16 WBS diagram illustrating the technical phases of the project . . . 61
A.1 Embeddings of the baseline LVMs on the rat hippocampus dataset. . . . 76
A.2 Embeddings of CEBRA on the rat hippocampus dataset. . . . 77
A.3 Embeddings of GP-VAE on the rat hippocampus dataset. . . . 78
A.4 Embeddings of the baseline LVMs on the macaque reaching dataset. . . . 78
A.5 Embeddings of CEBRA on the macaque reaching dataset. . . . 79
A.6 Embeddings of GP-VAE on the macaque reaching dataset. . . . 80
B.1 Violin plot of CEBRA-Dir across latent dimensions in the rat hippocampus benchmark. . . . 81

B.2 Violin plot of CEBRA-Pos across latent dimensions in the rat hippocampus benchmark. 81
B.3 Violin plot of CEBRA-PosDir across latent dimensions in the rat hippocampus benchmark. 82
B.4 Violin plot of CEBRA-Time across latent dimensions in the rat hippocampus benchmark. 82
B.5 Violin plot of CEBRA-Hybrid across latent dimensions in the rat hippocampus benchmark. 83
B.6 Violin plot of CEBRA-Dir across latent dimensions in the macaque reaching dataset. 83
B.7 Violin plot of CEBRA-Pos across latent dimensions in the macaque reaching dataset. 84
B.8 Violin plot of CEBRA-PosDir across latent dimensions in the macaque reaching dataset. 84
B.9 Violin plot of CEBRA-Time across latent dimensions in the macaque reaching dataset. 85
B.10 Violin plot of CEBRA-Hybrid across latent dimensions in the macaque reaching dataset. 85
B.11 Violin plot of GP-VAE across latent dimensions in the macaque reaching dataset. 86
C.1 Barplot of the best R^2 scores for global activity reconstruction of each model variety using linear estimators over latent dimensions for rat hippocampus dataset 87
C.2 Barplot of the best R^2 scores for positional decoding of each model variety using linear estimators over latent dimensions for rat hippocampus dataset 87
C.3 Barplot of the best R^2 scores for global activity reconstruction of each model variety using linear estimators over latent dimensions for macaque reaching dataset 88
C.4 Barplot of the best R^2 scores for positional decoding of each model variety using linear estimators over latent dimensions for macaque reaching dataset 88

List of Tables

1 Hyperparameters included in the benchmark analysis. Only hyperparameters that directly influence the learned latent representations are considered. 39
2 Comparison of baseline LVM methods optimal performance in rat hippocampus dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin). 45
3 Comparison of CEBRA methods optimal performance in rat hippocampus dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin). 46
4 Ranking of LVMs for the rat hippocampus benchmark based on positional decoding performance, global reconstruction performance and the Euclidean distance to the ideal point $(1, 1)$ in the two-objective space. . 50
5 Comparison of baseline LVM methods optimal performance in macaque reaching dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin). 53
6 Comparison of baseline LVM methods optimal performance in macaque reaching dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin). 54
7 Ranking of latent variable models for the macaque reaching benchmark using only latent dimensions, based on positional decoding performance, global reconstruction performance, and the Euclidean distance to the ideal point $(1, 1)$ in the two-objective space. 57
8 Work Breakdown Structure (WBS) activities, precedence relationships, and estimated durations for the project. 59
9 SWOT analysis 62
10 Summary project's budget 65

List of Abbreviations

MEA	Microelectrode Array
CaIm	Calcium Imaging
ML	Machine Learning
LVMs	Latent Variable Models
NP	Neuropixels Probes
BCI	Brain-Computer Interface
DR	Dimensionality Reduction
DL	Deep Learning
PCA	Principal Component Analysis
UMAP	Uniform Manifold Approximation and Projection
t-SNE	t-Distributed Stochastic Neighbor Embedding
CEBRA	Consistent Embeddings of Recordings using Auxiliary variables
AE	Autoencoder
VAE	Variational Autoencoder
GP	Gaussian Process
GP-VAE	Gaussian Process Prior Variational Autoencoder
AP	Action Potential
FA	Factor Analysis
ANNs	Artificial Neural Networks
KL Divergence	Kullback–Leibler Divergence
ELBO	Evidence Lower Bound
ICA	Independent Component Analysis
GRU	Gated Recurrent Unit
RNN	Recurrent Neural Network
InfoNCE	Information Noise-Contrastive Estimation

Contents

Abstract i

Resum ii

Acknowledgements iii

List of figures iv

List of tables vi

List of abbreviations vi

1 Introduction 1
- 1.1 Objective . . . 2
- 1.2 Methodology . . . 3
- 1.3 Scope . . . 4
- 1.4 Limitations . . . 5

2 Background 7
- 2.1 Neural Recordings . . . 7
  - 2.1.1 Electrophysiology Recordings . . . 8
  - 2.1.2 Calcium Imaging Recordings . . . 10
- 2.2 Mathematical Background . . . 10
  - 2.2.1 Mathematical Formulation . . . 12
  - 2.2.2 Neural Manifolds . . . 12
- 2.3 Machine Learning Foundations . . . 14
- 2.4 Latent Variable Models . . . 15
  - 2.4.1 Linear Approaches . . . 15
  - 2.4.2 Non-linear Approaches . . . 17
- 2.5 State of the art . . . 18
  - 2.5.1 Deep Generative Latent Variable Models . . . 18
  - 2.5.2 Contrastive Learning Approaches . . . 21
- 2.6 Market Analysis . . . 25

3 Concept engineering 27
- 3.1 Datasets . . . 27
  - 3.1.1 Rat Hippocampus Dataset . . . 28
  - 3.1.2 Macaque Reaching Dataset . . . 29
- 3.2 Model Implementation and Training . . . 30
  - 3.2.1 Baseline Models . . . 31
  - 3.2.2 Non-generative Deep LVM (CEBRA) . . . 31
  - 3.2.3 Generative Deep LVM (GP-VAE) . . . 32
- 3.3 Model Evaluation . . . 33
  - 3.3.1 Linear Regressor . . . 34
  - 3.3.2 Non-linear Regressor . . . 34
  - 3.3.3 Positional Decoding Analysis . . . 35

3.3.4 Global Activity Decoding Analysis 36

4 Detailed engineering **37**
4.1 Software Architecture 37
4.2 Benchmark Workflow 38
4.2.1 Baseline Hyperparameters 39
4.2.2 CEBRA Hyperparameters 39
4.2.3 GP-VAE Hyperparameters 40
4.2.4 Evaluation Framework 41

5 Results **43**
5.1 Rat Hippocampus Benchmark 43
5.1.1 Baseline Results 44
5.1.2 CEBRA Results 46
5.1.3 GP-VAE Results 47
5.1.4 Pareto Analysis 48
5.2 Macaque Reaching Benchmark 51
5.2.1 Baseline Results 51
5.2.2 CEBRA Results 53
5.2.3 GP-VAE Results 54
5.2.4 Pareto Analysis 55

6 Execution schedule **59**
6.1 Work Breakdown Structure (WBS) 59
6.2 Execution Chronogram (GANTT) 60
6.3 Critical Path Method Diagram (PERT) 60

7 Technical viability **62**

8 Economic viability **65**

9 Regulatory and legal aspects **66**

10 Conclusions and future outlook **68**

11 Bibliography **70**

A Appendix A **76**

B Appendix B **81**

C Appendix C **87**

1 Introduction

The origin of behavior arises from the highly complex activity of neuronal populations distributed across interconnected brain regions that act in concert to compute responses to relevant stimuli. Brain computations rely on the coordinated activity of neural populations in specialized regions, which continuously interact and integrate information [1], [2]. This highly organized network processes incoming information through dynamic neural activity, encoding relevant features into specific activity patterns that can be decoded and interpreted by other regions to guide meaningful decisions [3]. To better understand the computational capacity of brain circuits, neural dynamics during adaptive behavior are modeled through data-driven approaches involving statistically powerful machine learning (ML) techniques. Most common strategy in neural dynamics modeling involves the use of dimensionality reduction (DR) techniques to obtain low dimensional embeddings that reflect the coordinated neural dynamics correlated to given behavior in a more interpretable way [4].

In recent years, emerging advances in neural recording technologies have enabled access to the simultaneous activity of large neuronal populations at unprecedented cellular resolution [5]. Neuroscientists can now obtain highly dimensional and extremely complex recordings of neural activity across a variety of brain areas under specific experimental conditions. However, this rising complexity in recorded data requires requires computationally intensive tools capable of uncovering and interpreting the neural population dynamics underlying adaptive behavior [6].

Among several DR approaches, an emerging strategy is the use of latent variable models (LVMs). This modeling approach is an instantiation of DR techniques, which state that the observable data can be explained in a low-dimensional space of hidden latent factors that capture the underlying structure of data. In the context of neural recordings, LVMs aim to project high-dimensional recordings onto a low-dimensional latent spaces, extracting interpretable embeddings that capture shared population dynamics revealing the underlying correlated of behavior to neural activity [7], [8]. LVMs commonly shifts toward population-level analysis, implying that the computational capacity of the brain arises from the coordinated activity of large neuronal ensembles [4]. Therefore, LVMs are able to capture the activity states of these populations, encoded into coordinated spiking patterns within these neurons, while discarding the variance of independent neurons. Thus, neural states of populations encompassing hundreds of neurons are effectively summarized into interpretable embeddings, providing visual representations of these states into clusters [9].

LVMs emulate the brain's capacity by effectively encoding the coordinated activity states of neural populations into relevant features projected onto low-dimensional spaces[3]. These representations provide meaningful information about the stimuli that evoke the dynamical states of neural populations. Consequently, in most scenarios, LVMs performance is closely linked on their decoding capacity, which evaluates how the embedding points encoded from the neural activity actually correlate to the actual stimulus [3], [8]. The evaluation of these models mainly targets the quality of the produced embeddings by reconstructing certain stimulus or behavioral labels through specific decoders and

comparing the real stimulus using specific metrics [3].

However, more rigorous evaluation approaches investigate the extent to which latent embeddings preserve information about the neural activity from which they are derived. The objective is to quantify how strongly the learned representations remain correlated with the underlying population dynamics, thereby providing a quantitative measure of how faithfully the embeddings capture the structure of the neural manifold [3], [8]. This analysis is particularly relevant because the geometry of latent representations can be heavily influenced by behavioral supervision [9]. In such cases, embeddings may achieve excellent behavioral decoding performance while simultaneously discarding information about the original neural activity. As a result, strong decoding performance alone does not necessarily imply that the learned latent space accurately represents the neural processes that generated the data. Moreover, when the LVM is excessively optimized to match a specific target state space, it may suppress variability that is not directly related to the optimization objective. This process can attenuate emergent features and large-scale changes in neural activity. For example, if the overall mean activity of a neural population shifts because a subject transitions to a different arousal or behavioral state, the LVM encoding may remove or compress this global baseline change in order to preserve geometric alignment across behavioral conditions. Consequently, biologically meaningful information about neural signal can be lost, despite the apparent quality of the resulting embedding [10].

Currently, with the rise of deep LVMs, based on deep learning (DL), conventional linear approaches such as PCA, as well as other non-linear alternatives such as UMAP or t-SNE [7], [11], are increasingly being surpassed by DL-based approaches, as they can handle better large-scale datasets produced by neural recordings and capture more accurately the inherent non-linear complexity of neural dynamics [3]. Main deep LVMs approaches consist of unsupervised representation learning paradigms based on generative models, such as variational autoencoders, which enable to produce consistent embeddings of population activity accurately reflecting neural dynamics [12]. However, to that produce higher quality embeddings, recent deep LVMs are evolving toward self-supervised and supervised learning frameworks, in which auxiliary variables, including temporal indices and/or behavioral labels, also participate in the training process to produce the embeddings, leading to more robust, consistent representations that provide better performance in decoding tasks [8], [9].

1.1 Objective

The main objective of this project is to conduct a comprehensive benchmark of contemporary LVMs commonly employed for the analysis of *in vivo* neural recordings. The benchmark evaluates the ability to learn low-dimensional embeddings that accurately recover behavioral variables encoded by neural population dynamics while simultaneously preserving the structure and information content of the original neural activity. To this end, a multi-objective analysis is conducted to assess which LVMs achieve the best trade-off between positional decoding performance and preservation of the original neural signal under different experimental scenarios. Furthermore, the latent dimensionality associated with the optimal solutions is analyzed to evaluate how efficiently

each model compresses behaviorally relevant and neural information into compact representations.

A second objective is to investigate the influence of representative hyperparameters on the quality of the learned latent representations. Particular attention is given to hyperparameters that directly affect the encoding capacity of the models and, consequently, their performance in behavioral decoding and neural activity preservation tasks. Through a systematic exploration of hyperparameter combinations, the benchmark aims to identify the configurations that maximize the selected evaluation metrics and provide the best trade-off between competing objectives.

The project also examines the impact of supervision during representation learning. Specifically, it compares supervised frameworks, in which behavioral labels are explicitly incorporated into the training process, with conventional unsupervised approaches that rely solely on the structure of the neural recordings. This comparison aims to determine how supervision influences decoding performance, neural information preservation, and the geometry of the resulting latent manifolds.

Furthermore, the benchmark compares generative and non-generative LVMS strategies. By evaluating their respective abilities to represent neural population activity and capture behaviorally relevant information, the study seeks to characterize the strengths and limitations of each approach and to discuss their suitability for different neuroscience applications.

Consequently, the benchmark aims to identify the most promising LVMS frameworks for future neural decoding systems. By jointly considering behavioral decoding performance, preservation of neural dynamics, robustness across datasets and sensitivity to hyperparameter selection, the study provides insights into the ability of LVMs to capture accurate and biologically meaningful representations of neural activity.

1.2 Methodology

This exploratory analysis follows a methodological workflow divided into several stages.The first stage of the project consists of acquiring the theoretical and technical knowledge required for the study. This includes a review of the fundamental concepts of computational neuroscience, neural population dynamics, manifold learning and latent variable modeling. The objective of this phase is to establish the necessary background for implementing, training, and evaluating the LVMs included in the benchmark.

Once the theoretical foundations have been established, the two selected neural recording datasets are acquired and preprocessed. Subsequently, the LVMs are implemented and trained on the neural activity data. Depending on the learning framework, the models may also leverage auxiliary information associated with the recordings, including temporal indices, discrete contextual variables and continuous behavioral labels. These variables can used either during training for shaping the neural manifold, in case of supervised learning models.

The learned embeddings are evaluated through decoding analyses designed to quantify the amount of behavioral and neural information preserved in the latent representations.

For this purpose, two regression models are trained using the embeddings as inputs: a linear regressor and a nonlinear regressor. These decoders are trained to predict two continuous target variables: the positional label and the global activity metric derived from the neural population activity. Decoding performance is quantified using the coefficient of determination.

In addition to quantitative evaluation, the learned embeddings are visualized through low-dimensional plots to qualitatively assess their structure and their ability to capture behaviorally relevant information.

After verifying that all models operate correctly and produce reliable results, a systematic hyperparameter exploration is conducted to identify configurations that maximize decoding performance and neural information preservation.

Finally, to assess the versatility of the latent variable models across multiple objectives, a Pareto-front analysis is performed, allowing the identification of the LVMs that achieve the most favorable trade-offs between behavioral information recovery and neural signal preservation.

1.3 Scope

The scope of this project involves detailed analysis of the LVMs included in this benchmark. Specifically, this project encompasses mainly deep LVMs which operate through neural network architectures. Other more simple LVMs approaches participate in this benchmark, offering a baseline role for evaluation metrics. These baseline LVMs consists of classical DR methods such as principal component analysis (PCA), t distributed stochastic neighbor embedding (t-SNE) or uniform manifold approximation and projection (UMAP), providing the standard performance reference for assessing the minimal quality for more complex latent representations strategies. Conversely, deep LVMs include both non-generative and generative approaches for yielding neural representations. Due to their complexity, their hyperparameters are explored to assess their relevance in the LVMs performance.

The selected non-generative approach consist of CEBRA, a recently developed model that operates through contrastive learning to generate consistent embeddings without defining any generative process of the data [9]. This CEBRA framework is evaluated across all its available configurations, including variants trained using temporal indices, discrete and continuous variables. This allows a comprehensive assessment of how these auxiliary variables participate on shaping the structure of the learned embedding and how this influences decoding performance and signal preservation.

For generative modeling, the chosen model is an implementation of the conventional variational autoencoder, extensively used as a generative approach of LVMs, which includes gaussian process prior for shaping the latent space and a sequential network coupled to the encoder for capturing temporal relationships among neural signal. This generative LVM, referred as GP-VAE, was already developed in the master's thesis of Ros et al. [13]. For this work, with the purpose to equate this generative approach to the supervised framework of CEBRA, the training objective is further modified

by incorporating an additional contrastive term to enable the exploration of supervised learning paradigms within a generative modeling framework. This term is weighted by a dedicated hyperparameter, allowing control over the influence of contrastive supervision on the latent space organization.

The LVMs are evaluated on two electrophysiology neural recording datasets. The first one consists of different rat subjects freely exploring a linear track and the second involves proprioceptive feedback of a monkey holding a manipulandum that provides force bumps. Although the second dataset includes also motor task recordings, where the subject displaces the manipulandum to specific directions, these are intentionally excluded from the benchmark. This decision is motivated by the objective of isolating proprioceptive and spatial encoding properties within the neural representations, rather than disentangling multiple task modalities within the same neural recordings. In both cases, the primary evaluation target is positional decoding, focusing exclusively on proprioceptive or spatial information.

In addition, this project focuses on visualization of latent representations produced by the models, through specific plots of the embedding points computed by each model variety. Additionally, multiple plots of the evaluation metrics related to the hyperparameters are included for examining the influence in the quality of the embeddings.

All computational experiments are conducted on a GPU-equipped workstation at the Abante Lab in the Department of Biomedical Sciences. The datasets are publicly available through cloud storage repositories and are accessed via download links provided in the CEBRA framework [9]. All models used in this benchmark are publicly available and rely on existing implementations, allowing the focus of this work to remain on systematic evaluation rather than model development.

1.4 Limitations

Several limitations should be considered during the development of this project. Due to the tight timeline for project completion, only a limited number of relevant LVMs approaches are included in the benchmark, excluding a significant group of LVMs following different strategies.

In addition, the embedding analysis only focuses on decoding performance. Other forms of embedding analysis involving deeper mathematical or topological properties are beyond the scope of this project, although they could provide additional insights.

Model simulations also present practical constraints. Producing a sufficiently wide range of results across different parameter combinations requires extensive computational time, potentially scaling to several days. This can become problematic if errors occur during long simulation runs. Although the available computational resources are theoretically sufficient to complete the project, it is important to emphasize that the simulations remain computationally demanding. In addition, server architectures are prone to failures, which may contribute to delays in project completion.

Regarding the data source, the handling of available neural recordings also impose several constraints on the benchmark. First, the analysis is limited to single-session

recordings, which prevents to test consistency across different sessions. In addition, only electrophysiology recordings are considered, excluding other potentially modalities of high interest such as imaging or calcium-based approaches.

The number of recorded neurons is sufficient to conduct this benchmark analysis, but brain complexity arises from the contribution of multiple regions encompassing significantly more neurons. Therefore, the LVM's performance is constraint by the population size. The inferred latent factors for this partial measures of brain's computations represent only an approximation of the collective activity of the full neural population. These limitations may prevent models from fully capturing the complex dynamics of neural circuits and the underlying principles governing neural state transitions.

2 Background

2.1 Neural Recordings

To elucidate how neurons interact to process information and elicit responses, neuroscience increasingly focuses on the coordinated activity of neural populations rather than individual units [4]. Typical procedure involves simultaneous recording high-resolution electrical or optical signals from large groups of neurons distributed across brain regions based on their functional roles [5]. As the brain is a highly interconnected organ composed of multiple functional regions, contributing in varying degrees to the encoding of a given stimulus, neural recordings should encompass sufficient spatial coverage and neuronal sampling to capture this encoding capacity [3]. Increasing the number of recorded neurons enables a more accurate and comprehensive readout of neural computations [5]. This facilitates a more faithful characterization of the population's dynamical states under specific behavior, yielding latent representations that better reflect the underlying neural encoding processes. Consequently, incorporating richer and more extensive neural data enhances decoding performance [8].

In case of LVMs, neural recordings must maintain single-cell resolution while including large groups of cells because these models rely on patterns of covariation across populations to reveal the internal states that drive neural computation capacity [8]. Each individual neuron provides a different view of the underlying shared neural process, contributing to a more accurate modeling of neural dynamics [4], [14]. Single-cell resolution allows models to capture more nuanced patterns of activity, enabling better generalization to new behaviors and thus improving decoding performance [15].

For the purposes of this thesis, population-level analysis is preferred over individual-level alternatives because relationships between brain and behavior often remain elusive when viewed through the perspective of single independent units [4], [16]. Individual neurons typically yield highly heterogeneous and complex responses that may bear no obvious relationship with sensory inputs or motor outputs, requiring context of surrounding neural signaling to understand the encoding processes arising behavior [3]. In contrast, population-level analysis which cover neural mechanisms emerging from collective patterns inaccessible through independent single-neuron firing rates [4], [17]. Nevertheless, studying neuronal activity from a single-cell perspective is interesting in other applications, such as multimodal single-cell integration, allowing researchers to integrate functional and molecular readouts, for example [18].

Contemporary neuroscience is witnessing a paradigm shift from recording individual units to monitoring vast neuronal populations simultaneously [5]. Modern high-density interfaces now enable the acquisition of well-isolated neural activity from hundreds to thousands of neurons across multiple brain regions [5], [19], [20]. These large-scale single-cell resolution recordings are essential for capturing high-dimensional dynamical portraits of neural circuit computations, providing underlying information about sensory and cognitive functions [5].

Within electrical signals arising from neurons, action potentials (APs) provide the most critical insights into neural communication mechanisms, as they serve as the primary

vehicle for signal transmission within neural circuits [21]. An AP is a biophysical event occurring when a neuron's transmembrane potential reaches a specific threshold. These events are recorded as spikes which typically manifest as high-frequency signals exceeding a background noise threshold [22]. The activity of each neuron is represented as a time-varying spiking profile. The number of spikes per unit time, defined as the firing rate, is used to quantify neuronal activity. [4]. Recent technological advancements have significantly reduced the invasiveness of these measurements while improving their long-term reliability, enabling stable *in vivo* recordings in freely moving animals without significant signal degradation [19]. In contrast, *in vitro* neural recordings provide an experimental framework for investigating specific properties of neuronal populations, such as plasticity or primary responses to treatments [23]. However, within the context of this thesis, where the primary objective is to study the relationship between behavior and brain activity, *in vivo* recordings are preferred over *in vitro* approaches because they provide behavioral context.

A single experiment for *in vivo* single-neuron recordings can follow the same neural population across multiple sessions spanning days or even weeks [9], [24], [25]. This longitudinal approach is critical for investigating how neural circuits evolve during fundamental biological processes such as learning, development or aging [8], [16].

Experimental data collection is typically organized in a hierarchical structure of sessions and trials. Each session is composed of multiple trials representing repeated instances of specific experimental conditions [8]. This trial-structured design allows researchers to leverage statistical power across repetitions to extract de-noised population-level representations [4].

2.1.1 Electrophysiology Recordings

Electrophysiology recordings allow the direct measurement of the electrical fields generated by neuronal signaling by transducing neural activity signals into recorded digital forms [25]. Among the available techniques for monitoring neuronal electrical activity, intracellular and extracellular recording approaches are particularly prominent due to their distinct characteristics and applications. Although intracellular recordings provide high-resolution information on subthreshold synaptic inputs, they remain technically demanding and inherently invasive. This invasiveness poses a substantial risk of damaging the cell membrane, thereby limiting their suitability for large-scale population recordings [22], [25]. Consequently, extracellular approaches are often preferred because they permit minimally invasive, long-term monitoring of neural populations. In addition, most extracellular techniques provide sufficient spatial and temporal resolution to capture relevant extracellular signals without compromising plasma membrane integrity [25], [26].

The implementation of high-density research platforms often utilizes microelectrode arrays (MEAs), which consist of multiple micro-sensors designed to read variations in voltage within the extracellular space [27]. These electrodes detect extracellular field potentials, ranging from tens to hundreds of millivolts, caused by the current flows of all ionic processes across nearby neurons, enabling the simultaneous activity detection of several neurons. In these systems, the extracellular space acts as a volume conductor

where contributions from various neuronal unit sources sum linearly, with their signal magnitude being weighted inversely to their distance from the recording site [22]. A significant challenge in these extracellular recording strategies is that the raw signal typically originates from a multitude of nearby neurons, resulting in a convolved voltage trace [28]. To delineate the functional contributions of individual cells within a neural circuit, the aggregate extracellular signal must be decomposed into discrete spiking profiles via a process known as spike sorting [29]. This analytical pipeline typically begins with band-pass filtering raw voltage traces to isolate high-frequency activity, followed by the detection of discrete events using amplitude thresholding. These detected spikes are subsequently categorized into specific units by clustering waveforms based on their unique geometric shapes and features. This classification is grounded in the fact that each neuron possesses a distinct morphology and maintains a specific spatial distance and orientation relative to the recording sensor, resulting in a characteristic and identifiable extracellular footprint [22], [28].

High-density MEAs facilitate single-unit isolation by capturing the activity of an individual neuron across multiple adjacent recording sites. Nonetheless, spike sorting remains a challenging and imperfect process, typically requiring automated algorithms followed by rigorous manual curation to ensure tracking accuracy [29]. Achieving successful single-unit isolation is a critical prerequisite for subsequent functional analyses, as it enables the precise reconstruction of temporal dynamics and synchronization patterns within neural populations [8]. Beyond algorithmic constraints, material and anatomical challenges must be also considered. The implantation of rigid microelectrodes into living brain tissue invariably triggers a foreign body inflammatory response. Furthermore, recording from deep-brain structures, such as the hippocampus or thalamus, requires needle-like penetrating probes rather than planar arrays [30]. Consequently, conventional planar MEAs have historically been preferred for *in vitro* applications rather than *in vivo* preparation due to these acute biophysical limitations. However, recent advancements in MEA technology have significantly reduced probe dimensions and structural invasiveness while enhancing deep-tissue accessibility, thereby increasing the feasibility and reliability of in vivo high-density recordings [31].

Neuropixels probes (NP) are relatively recent extracellular electrophysiology which allows for large-scale monitoring of neural population dynamics at single-neuron resolution *in vivo*, allowing the simultaneous recording of hundreds of neurons within a reduced space while improving data quality and minimizing invasiveness. Each probe consists of a single long shank with a microscopic cross-section, densely populated with recording sites, enabling the simultaneous capture of activity from hundreds of neurons across both superficial and deep brain structures. This architecture minimizes tissue damage and displacement while providing the unprecedented spatiotemporal resolution required for brain-wide recordings [32]. NP are currently among the most advanced options for electrophysiology recordings, offering increased spatial coverage along with a substantial improvement in neuronal yield and single-unit isolation compared to their predecessors [19], [25]. These advantages arise from their high channel count and dense arrangement of recording sites, which enable brain-wide recordings at single-cell resolution, spanning both superficial and deep structures. Additionally, they provide high signal fidelity with minimal interference from external noise sources [32].

2.1.2 Calcium Imaging Recordings

Optical recordings based on calcium imaging (CaIm) facilitate the monitoring of large neural populations with significantly greater coverage and lower invasiveness than conventional electrophysiology [25]. This methodology relies on the biological principle that neuronal AP trigger a brief, millisecond-scale influx of intracellular calcium. These events are detected by staining cells with activity-dependent calcium indicators, which modulate their fluorescence intensity in response to these transients [33].

CaIm conventional procedure involves the use of two-photon microscopy, which provides depth penetration of several hundred microns into intact tissue, enabling high-resolution functional imaging of local networks in behaving animals [20]. Compared to penetrating microelectrodes approaches, which can cause tissue displacement and trigger chronic immune responses that lead to recording instability, CaIm through cranial windows or thinned skulls is minimally invasive and allows for the durable, long-term tracking of the same individual neurons over weeks or even months [33].

Despite their high spatial coverage, optical imaging modalities possess inherent limitations, particularly regarding temporal resolution [32]. Because the biophysical kinetics of intracellular calcium and its corresponding fluorescent indicators are significantly slower than the millisecond-scale dynamics of neuronal firing, CaIm provides only an approximation for actual spike timing and rates [25], [33]. To estimate the underlying electrical activity, mathematical spike deconvolution techniques are mandatory to infer discrete action potential timings from these slow fluorescence transients. Consequently, the fidelity of subsequent analyses is heavily contingent upon the efficacy of these deconvolution algorithms. An improper implementation or flawed modeling assumptions may introduce significant biases into downstream data interpretation[32], [34]. Furthermore, signal accuracy can be also hindered by contamination agents, where background fluorescence from surrounding axons and dendrites obscures the activity of the targeted cell bodies [33].

In the current landscape of systems neuroscience, CaIm and NP have emerged as the dual gold standards for neural recording [8]. These technologies enable the acquisition of massive datasets at an unprecedented scale, providing the empirical population-level foundation for LVMs [9]. CaIm is distinguished by its extensive spatial coverage and resolution, with recent advancements allowing for the simultaneous monitoring of up to one million neurons [33]. In contrast, NP provide a superior capacity to capture rapid spiking dynamics and local field potentials with sub-millisecond temporal resolution across multiple superficial and deep brain structures [32]. Following a trajectory of exponential growth in recording capacity, both modalities are expected to continually evolve, further optimizing the quality and quantity of data acquisition to facilitate brain-wide insights into neural function [19], [20].

2.2 Mathematical Background

Despite significant technological progress in recording modalities, the inherent complexity of neural signals still presents a substantial barrier to interpretation [5]. The resulting data often appear as a confounding collection of highly heterogeneous and

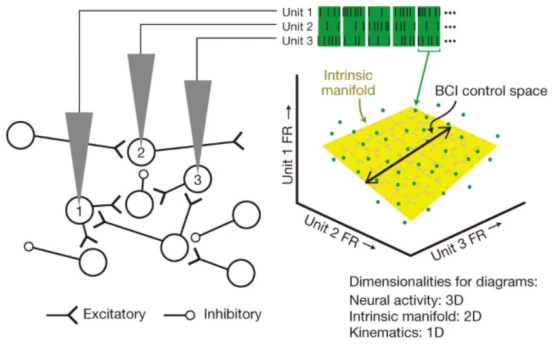

Figure 1: The evolution of population dynamics from neural recordings are constrained within a manifold embedded in a latent space. Schematic by Sadtler et al. [37]

noisy responses with any apparent structure [4]. To resolve this complexity, researchers leverage abstract mathematical frameworks, heavily relying on population-level analyses, for transforming these uninterpretable raw signals into consistent and identifiable representations based on collective dynamics, providing insights of the encoded information [35].

Neurons in large populations do not act independently but instead exhibit coordinated fluctuations and redundant patterns of activity [35]. Their activity is characterized by continuous network co-modulation, driven by a sophisticated interplay of excitatory and inhibitory signals. By characterizing patterns of covariation across the population, abstract mathematical methods can distill high-dimensional recordings into a low-dimensional, interpretable representations capturing the possible collective states of the neural circuit [36].

Most mathematical tools for analyzing neural recordings focus on population activity, represented as time-varying firing rate signals. These signals are derived from the raw spike trains of recorded neurons over time. The processing typically begins with temporal binning, in which the recording session is partitioned into short, fixed-length intervals. Each bin contains the number of spikes that occur within its time window, transforming the raw spike trains into discrete spike counts over time. In some cases, to facilitate further analysis, these discrete signals are then convolved with a temporal filter, yielding a continuous estimate of each neuron's time-varying firing rate [8]. This approach omits single-spike variability and reveals the continuous evolution of population activity [16]. However, discrete spiking data is also utilized instead to preserve information about the original signal.

To resolve this complexity, observable, high-dimensional neural data is assumed that can be embedded into a latent, lower-dimensional space. Consequently, neural recordings produced by large populations of neurons can be conceived as projections of underlying latent variables that characterize the true internal state of the biological network. Under this assumption, all co-modulation patterns can be effectively described by latent representations residing in a intrinsically lower dimensional subspace, which provides interpretable portraits of the dynamics that drive behavior [8], [35].

2.2.1 Mathematical Formulation

Given a recorded population of N neurons across T time points, the neural or ambient state space $x(t)$ is defined as an N-dimensional Euclidean coordinate system where each axis represents the time-varying firing rate of a single constituent neuron [4]. Within this framework, the collective activity of the population at any instantaneous time point t is represented as a single N-dimensional population vector or neural state:

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_N(t) \end{pmatrix} \tag{1}$$

As time progresses and the subject engages in certain task or stimulus, the patterns of activity evolve through the modulation of individual firing rates, causing the neural state to trace a continuous neural trajectory through the ambient space [16]. While this high-dimensional space mathematically encompasses all possible combinations of firing rates within the population, empirical evidence demonstrates that the system does not explore all available points, but only a subset, indicating that the collective activity of neurons is constrained [35], [38]. This observation supports the theory that high-dimensional observable signal is a projection of a few underlying latent variables that characterize the true internal state of the biological circuit. The population states are constrained in a structure within this latent space, referred to as the neural manifold [35]. In neuroscience, the intrinsic neural manifold is defined as the low-dimensional surface that captures the possible collective states [16], [37], [39].

Consequently, neural activity arises from an inherent D-dimensional subspace (where $D << N$), containing the lowest number of dimensions to define the neural manifold that constraints the population activity $x(t) \in \mathbb{R}^N$. Therefore, lower dimensional representations $z(t) \in \mathbb{R}^D$ can be used to effectively describe population dynamics in a more interpretable way [4].

$$z(t) = \begin{pmatrix} z_1(t) \\ z_2(t) \\ \vdots \\ z_D(t) \end{pmatrix} \tag{2}$$

2.2.2 Neural Manifolds

Mathematically, a manifold is defined as a local Euclidean topological space. which implies that it appears approximately flat at a local scale, even with complex shapes

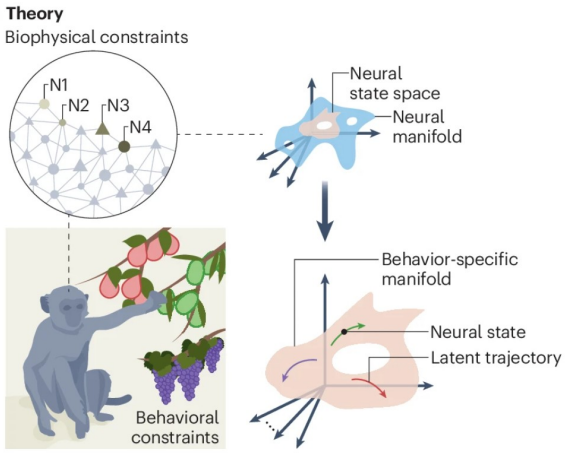

Figure 2: Biophysical properties and processes constraints the population states into a neural manifolds. Schematic by Perich et al. [16]

[40].

Following the definition of neural manifold, the latent representations from recorded data lie on a low-dimensional manifold that imposes constraints on the system, delimiting the set of allowable joint activity patterns, as illustrated in Figure 1. Furthermore, the manifold coordinates the temporal evolution of the system, as continuous transitions between distant neural states are constrained to move through a sequence of adjacent, intermediate states along the manifold's surface [38], [39].

The central theoretical assumption of this framework is that the geometric and topological properties of this shape provide profound insights into the mechanisms of neural circuits and the nature of the variables encoded within them [35]. All the possible collective states defined by the manifold's shape emerge from a combination of intrinsic biophysical constraints and extrinsic behavioral constraints [16] as illustrated in Figure 2. Intrinsic constraints arise from the properties of the network itself, including connectivity, synaptic strengths, cell-type distributions and activity of glia cells, all of which delimit the set of allowable joint activity patterns. Conversely, extrinsic constraints are imposed by task structure, sensory inputs and behavioral demands, which together determine the portion of the manifold explored by neural activity [37].

The existing structure of a neural manifold acts as a causal constraint on learning and plasticity. Research demonstrates that within-manifold perturbations, which require new associations between existing co-modulation patterns, are readily learned on short timescales [37]. In contrast, outside-manifold perturbations are significantly harder to acquire because they require the generation of entirely new neural activity patterns and connections, causing the expansion and re-orientation of the manifold itself [37]. Over longer periods, learning can create *de novo* geometric representations for the manifold [41]. Consequently, extracting the neural manifold can be valuable for identifying the constraints governing neural population activity and understanding how these dynamics evolve through topological and geometric analyses.

2.3 Machine Learning Foundations

Machine learning (ML) is an interdisciplinary field concerned with algorithms that can learn patterns from data and adapt their behavior accordingly [42]. In the context of this thesis, only data-driven and task-driven ML approaches are included, for model neural activity and predict contextual variables from learned representations, respectively [3].

In most ML models, learning is achieved by optimizing a set of parameters, typically including weights and biases, which define the mapping from inputs to outputs. These parameters are adjusted during training to minimize a predefined objective function, known as the loss function, which quantifies the discrepancy between the model predictions and the target values, thereby defining the optimization objective that the model seeks to improve over successive iterations [42].

Training is commonly performed using mini-batch optimization, where the full dataset is divided into smaller subsets, called batches [43]. An epoch corresponds to a full pass over the entire dataset, during which all batches are processed once. For each batch, the model computes the average loss and updates its parameters accordingly. The standard optimization procedure is gradient descent, where parameters are iteratively updated in the direction that minimizes the loss, as given by the gradient of the loss with respect to the model parameters. The magnitude of these parameter updates is controlled by a hyperparameter known as the learning rate α. The update rule is given by:

$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta f(\theta_t) \tag{3}$$

where θ_t denotes the model parameters at iteration t (typically corresponding to a single mini-batch update), and $f(\theta_t)$ represents the loss function evaluated on the current batch.

This optimization procedure is the standard training paradigm for most machine learning models. It typically relies on three data splits: training, validation and test sets. The training set is used to update the model parameters via gradient-based optimization. In contrast, the validation and test sets are not used for parameter updates [42].

The validation set is used during training to tune hyperparameters and to monitor generalization performance. It also plays a key role in early stopping, a regularization strategy used to mitigate overfitting. Overfitting occurs when a model learns patterns specific to the training data that do not generalize well to unseen samples. Early stopping monitors the validation loss, which is computed without updating model parameters. If the validation loss does not improve for a predefined number of consecutive evaluations (controlled by a patience parameter), training is terminated to prevent further overfitting [44].The test set is reserved exclusively for final model evaluation, providing an unbiased estimate of performance on unseen data.

This thesis encompasses both supervised and unsupervised learning approaches. In supervised learning, the model learns patterns in the data that are associated with an external variable, which is used to guide the training process. In contrast, unsupervised learning relies solely on the structure of the input data to learn intrinsic patterns, with-

out access to external labels or auxiliary variables. In addition to these two paradigms, more specialized frameworks are also considered, such as self-supervised learning. Self-supervised learning can be viewed as a subclass of unsupervised learning in which the model generates its own supervisory signals from the data itself. In the context of this project, these signals are constructed using temporal indices, enabling the model to capture the sequential and chronological structure of neural recordings [45].

2.4 Latent Variable Models

Under neural manifold framework, in which observable activity coming from neural populations can be defined by low-dimensional manifolds, researchers utilize sophisticated mathematical tools consisting of LVMs to extract insightful representations [3]. These models compress high-dimensional neural data into low-dimensional embeddings points, which should cluster according to the neural manifold shape that constraints the recorded firing rates [4]. While linear methods are frequently used to increase interpretability through simple approximations, recent advancements leverage generalizations based on complex DL architectures to capture the inherent nonlinearities and dynamics of the circuit [46]. Despite their increased complexity and often reduced interpretability, deep LVMs can yield more reliable representations for decoding tasks, as they learn embeddings that lie on manifolds more accurately reflecting the constraints of neural population dynamics [9].

It is important to note that the true latent manifold capturing all the states of neural populations can not be exactly deduced, as LVMs heavily rely on assumptions for producing embeddings, and without a complete understanding of the brain's actual encoding strategies, these assumptions remain approximations of the true manifold [8]. Therefore, LVMs' encoding mechanism should aim to preserve the actual structure of data and minimize the assumptions to better approximate to the true neural manifold [3]. Other aspect to consider, specially in deep LVMs, is the lack of identifiability while producing latent representations from observable recordings. Identifiability is the desirable property that model parameters, in this case the latent factors, can be uniquely determined, ensuring that the learned representations are consistent [9]. Non-identifiable models might construct different latent representations for the same observed data distribution, leading to inconsistent embeddings across different animals, recording sessions or even repeated runs of the same algorithm on the same data [9]. By ensuring that an algorithm yields the same latent space given the same underlying data distribution, researchers can better address this neural activity correspondence problem [9]. In summary, LVMs should prioritize the learning of robust and stable latent representations that capture the range of neural states while closely approximating the true manifold.

2.4.1 Linear Approaches

Linear LVMs extract latent representations based on the variation of the firing rate patterns of the neurons, yielding embeddings that capture the linearly coordinated states of the population [4]. For this, these methods operate under the assumption of linearity, where the observed population activity vector $x(t)$ is modeled as a projection

from a lower-dimensional latent vector $z(t)$ through a linear mapping matrix G. Thus, the observed high-dimensional activity arises from linear combination of underlying latent signals [46]. In this framework, the latent factors represent characteristic patterns of co-variation that serve as linear approximations of the underlying circuit dynamics:

$$x(t) = G \cdot z(t) \tag{4}$$

Linear LVMs include several methods, such as Principal Component Analysis (PCA) and Factor Analysis (FA), which enable the extraction of meaningful representations through linear transformations based on variance structure [4]. In many studies, PCA is preferred as the primary linear LV model due its ease of implementation and the interpretability of the resulting embeddings. It operates by mapping a given input onto uncorrelated variables known as principal components, which correspond to the latent factors generating the observations. The procedure starts by extracting a given input $x(t)$, centering the data and computing the covariance matrix.

To this end, PCA begins by centering the data $\hat{x}(t) = x(t) - \bar{x}(t)$, where $\bar{x}(t)$ represents the mean of neural activity at a given instant t, and computing the covariance matrix:

$$\Sigma = \frac{1}{T}\hat{x}\hat{x}^T \in \mathbb{R}^{N \times N} \tag{5}$$

This symmetric matrix contains the patterns of variation of the original features . Then, singular value decomposition is applied, obtaining

$$\Sigma = Q\Lambda Q^T \tag{6}$$

where Q is the matrix of orthonormal eigenvectors and Λ is the diagonal matrix of corresponding eigenvalues. Each eigenvector defines a direction in latent space, and its eigenvalue the variance of the data along that direction. Accordingly, principal components are the eigenvectors with the largest eigenvalues, and projecting onto them yields a lower-dimensional subspace that preserves the maximum possible variance of the original data [47]. PCs are ordered by the amount of variance they capture. The first principal component is the direction along which the data has maximum variance, with each subsequent component capturing the largest remaining variance subject to being orthogonal to the previous ones. By selecting the top D components, we obtain a lower-dimensional representation in which the projected features are uncorrelated and retain most of the total variance of the original data. The goal is to keep enough components to explain a large fraction of the variance, rather than all of it, thereby reducing dimensionality while preserving the most significant structure in the data [47].

Thus, PCA can be employed as a simple yet effective baseline LVM, providing a useful reference for comparing the performance of more complex models and offering initial insights into how population activity organizes within a low-dimensional subspace [4].

However, neural dynamics typically exhibit nonlinear dependencies that cannot be adequately captured by linear LVMs [46]. True neural manifolds often possess complex,

curved topologies that are not well described by linear reconstructions, which are inherently limited to planar approximations. Consequently, linear methods may overestimate the intrinsic dimensionality of the manifold, requiring a larger number of Euclidean embedding dimensions to approximate the geometry of curved or warped structures [16]. Another limitation of linear approaches concerns their lack of consistency, as they rely exclusively on variance-ranked components that are specific to the recorded subset of neurons. These variance patterns can be highly sensitive to experimental conditions and batch effects, potentially obscuring the underlying neural dynamics and leading to inconsistent embeddings [46].

2.4.2 Non-linear Approaches

Linear LVMs fall short on trying to recover the true manifold, requiring to level up the complexity of the models to have decent embeddings [46]. Because biological circuits involve nonlinear neuronal responses and intricate recurrent connections that amplify these effects, neural population activity is governed by patterns that are often not accessible through conventional linear methods [46].

These nonlinearities emerge heterogeneously across brain regions, reflecting the brain's organization into functionally specialized areas that encode and decode different types of stimuli and information [36]. As a result, the corresponding neural manifolds exhibit complex geometries that can vary substantially across regions.

Moreover, manifold nonlinearities become more pronounced during complex tasks that require a broader repertoire of neural activity patterns [39]. In simpler tasks, which predominantly engage a limited number of regions while others contribute minimally, neural activity may be confined to a restricted portion of the ambient space, allowing linear approximations to provide reasonably accurate local estimates of the underlying manifold. However, as task complexity increases, a larger number of neural populations contribute to process the stimuli and the associated encoding patterns become more intricate. Consequently, the underlying manifold exhibits greater curvature and nonlinear structure in order to fully represent the neural states required to perform the task [46].

Nonlinear LVMs adopt a more realistic framework for describing neural manifolds by extending beyond the constraints of linear methods to learn representations capable of capturing complex, nonlinear patterns across neural populations. These models are able to recover more sophisticated geometrical structures rather than simple linear subspaces [37]. Such complex geometries enable a more precise mapping between population-level neural states and behavioral variables, thereby facilitating a deeper understanding of how specific neural ensembles support flexible behavior and function [46].

Nowadays, there exists a wide range of nonlinear approaches for LVMs, spanning from relatively simple methods capable of capturing nonlinearities in neural representations with low computational and temporal cost, to deep LV frameworks based on highly complex artificial neural networks (ANNs) [4], [9], [11], [48]. These deep models can capture richer and more intricate activity patterns, leading to more expressive and accurate manifolds.

Among the simplest nonlinear alternatives of LV modeling, t-SNE is is a widely used DR technique for the visualization of high-dimensional datasets, enabling to produce low-dimensional embeddings that preserve local neighborhood structure. [48]. The method operates by converting pairwise relationships between data points in the high-dimensional space into probability distributions, where similar points are assigned higher probabilities of being neighbors and dissimilar points receive lower probabilities. These similarities are typically computed from Euclidean distances. t-SNE then defines a corresponding probability distribution in the low-dimensional embedding space, commonly using a Student's t-distribution to better separate distant points. The embedding is optimized such that the low-dimensional distribution closely matches the high-dimensional one, thereby preserving local similarities between data points [48].

Another widely used non-linear LVM technique is UMAP, which shares conceptual similarities with t-SNE in that both methods aim to preserve local neighborhood relationships when constructing low-dimensional embeddings. However, UMAP explicitly incorporates attractive and repulsive forces during the optimization of the latent space, helping to separate distant clusters while maintaining meaningful local relationships. As a result, UMAP often preserves more of the global geometry of the data, scales more efficiently to large datasets, and typically produces embeddings with improved continuity and cluster organization [11].

2.5 State of the art

The modern era of neuroscience has been profoundly shaped by a data revolution, in which the ability to simultaneously record the activity of hundreds or thousands of neurons has driven a paradigm shift toward mathematical architectures capable of managing and analyzing such large-scale datasets [5]. In response, LVMs have increasingly evolved into deep generalizations through the incorporation of DL architectures, which are better suited to capturing the intrinsic nonlinearities underlying neural population dynamics. These data-driven approaches leverage high statistical capacity to extract more reliable, generalizable and robust latent representations [3].

2.5.1 Deep Generative Latent Variable Models

Deep generative LVMs explicitly parameterize the data-generating process, which allows for more interpretable and identifiable embeddings [13]. Commonly, these models consist of variation of variational autoencoders (VAEs), which are generative extensions of autoencoders (AEs). Traditional AEs can learn the complex patterns governing neural dynamics and yield low-dimensional representations preserving their inherent structure through two components: the encoder and the decoder. The encoder extracts the low-dimensional representations by mapping the input $x(t)$ onto the latent space $z(t)$, while the decoder participates in the inverse process by reconstructing the input $\hat{x}(t)$ from their encoded representation. AEs are trained to generate embeddings whose decoded reconstructions closely match the original input signal by minimizing a reconstruction loss. However, standard AEs are not inherently generative models, as they do not explicitly learn the underlying probability distribution of the data. Consequently, they may be prone to overfitting and often reduce the latent representation to deterministic

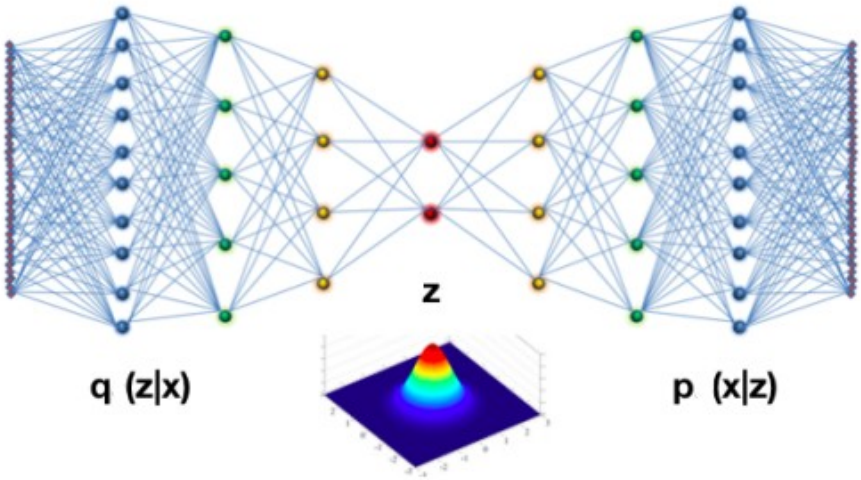

Figure 3: Variational autoencoders encoder-decoder structure. Figure from Odaibo et al. [50]

point estimates. This can result in embeddings that lack continuity, coherence, and an explicit representation of the uncertainty inherent to neural signals [49].

The incorporation of a probabilistic layer to AEs aims to address these limitations. VAEs learn the parameters of probability distributions governing both latent representations and observations, rather than independently encoding and decoding deterministic points [51]. The network architecture maintains the encoder-decoder structure, as shown in Figure 3. In particular, VAEs provide a probabilistic description of the data-generating process where the observations X and the latent factors Z can be modeled as random variables, jointly distributed by $p_\theta(X, Z)$ which configures the model defined by the set of parameters θ. Under a Bayesian framework, the observations arising from X can be described as the evidence $X \sim p_\theta(x)$ and the latent variable Z corresponds to the prior distribution $Z \sim p(z)$. Commonly, the prior is modeled using a standard normal distribution $p_\theta(z) = \mathcal{N}(0, 1)$ if there are not any strong beliefs or constraints about the latent space. The encoder aims to learn the posterior distribution $p_\theta(z \mid x)$ which relates how latent variables arise from the observations, while the decoder learns the likelihood $p_\theta(x \mid z)$ of this observed data under specific latent distributions.

$$p_\theta(z \mid x) = \frac{p_\theta(x \mid z)}{p_\theta(x)} p(z) \tag{7}$$

Ultimately, the model aims to yield accurate representations from the observations that encode their generative basis. Therefore, reconstruction capability of the model must be evaluated to ensure that the produced embeddings are actually linked to the underlying structure of the data [51]. In this probabilistic framework, computing the marginal distribution of the observations is computationally unfeasible as it is required to integrate over non-linear latent variables with non-analytical solutions. Consequently, it is usually impossible to derive the posterior distribution analytically, as it depends on the marginal as shown in Eq. (7). To overcome this limitation, the encoder learns

$q_\phi(z \mid x)$, which ultimately approximates to the posterior distribution of the model $q_\phi(z \mid x) \sim p_\theta(z \mid x)$.

$$p_\theta(x) = \int p_\theta(x, z)dz = \int p_\theta(x \mid z)p(z)dz \tag{8}$$

The similarity between a pair of distributions can be quantified using the Kullback–Leibler (KL) divergence. This metric is always non-negative and equals zero only when the two distributions are identical [52]. Therefore, KL divergence may serve as the optimization objective to approximate the learned encoded distribution to the posterior through its minimization. Thus, VAEs are commonly optimized through the evidence lower bound (ELBO):

$$\mathcal{L}_{\theta,\phi} = -\mathbb{E}_{q_\phi(z|x)}[\log(p_\theta(x \mid z)] + \beta D_{KL}(q_\phi(z \mid x) \mid\mid p_\theta(z)) \tag{9}$$

This loss function, corresponding to the generalization of a VAE usually referred to as β-VAE, can be conceived as the union of a reconstruction term and a regularization term. The reconstruction term $-\mathbb{E}_{q_\phi(z|x)}[\log(p_\theta(x \mid z)]$ targets the generative capacity of the embeddings by measuring how accurately the decoder can reconstruct the input data from the latent representation [50]. The maximization of this term encourages the model to generate more similar data from encoded representations, which is equivalent to minimizing the reconstruction error. The regularization term acts as a constraint for shaping the encoder distribution $q_\phi(z \mid x)$ according to the imposed prior $p(z)$, promoting continuity and structure in the latent space and improving the generative capabilities of the model [51]. The hyperparameter β adjusts the relative contribution of the regularization term, allowing to balance between reconstruction quality and latent space regularization [53].

However, to ensure the continuity and consistency of the produced embeddings' latent space regarding the original data, one common strategy is to guide the encoding process with some contextual information about the neural signal. The Gaussian process prior VAE (GP-VAE) was designed to enhance the cohesion of embedding representations by building a prior distribution such that all the latent representations over time follow a Gaussian process [54]. A Gaussian process is defined as a collection of random variables such that every finite collection of those variables has a multivariate normal distribution [55]

$$z_1, ..z_T \sim \mathcal{GP}(0, \mathcal{K}) \tag{10}$$

The latent variables are jointly gaussian with a covariance matrix $\mathcal{K}$, which contains the values of correlation between every pair of time instants. To maintain the sequential coherence, nearby points are imposed to have higher correlations, while temporally further points get lower correlations. To model this pattern of correlations based on temporal proximity, a kernel function consisting of radial basis function (RBF) is implemented. RBF models the temporal correlation under a gaussian imposition, using a

lengthscale σ and a scale parameter α to tune the shape of the kernel, allowing to adjust temporal sensitivity of the model in a flexible manner and capture dynamics at different timescales. To ensure that the GP captures well the desired dynamics, the optimal values of these parameters are learned during training to maximize the likelihood.

In addition, these models also allow for the introduction of inductive biases through neural network architecture choices, allowing models to further leverage temporal correlations among data-points. In recent work, Ros et al. proposed an approach where the encoder projects a sliding window in the latent space, instead of a single time-point[13]. To assimilate this kind of sequential inputs, the authors proposed a first layer in the encoder includes a gated recurrent unit (GRU), a type of recurrent neural network (RNN) able to handle temporal data [56]. The GRU layer participates on selectively retaining relevant information over the sequence, allowing to effectively extract temporal correlations for the encoder. The core of GRU are the hidden state layers H_t, which contain all the information learned up to the given time point, effectively acting as the network's memory. For each time-step, the corresponding hidden state receives information about the input at that time-step X_t and the hidden state of the previous time-step H_{t-1}. The model generates a candidate hidden state incorporating new information while retaining useful information from the learned context. The resulting hidden state corresponds to the update of the previous hidden state with candidate information. This updating is controlled by a two gating mechanism:

- **Reset Gate**: Filters out irrelevant information of the previous hidden state for the candidate
- **Update Gate**: Controls the trade-off between the previous hidden state and the candidate

2.5.2 Contrastive Learning Approaches

Generative models often rely on restrictive explicit assumptions about the underlying statistics of the data, which may not perfectly match the brain's actual encoding strategies. A critical weakness of many traditional deep generative LVMs is the lack of identifiability, leading to inconsistent embeddings that diverge when the algorithm is run repeatedly or applied across different subjects, as the model struggles to reconstruct the a highly complex generative process that produces the observed neural signal [9]. To overcome these limitations, some researchers explored the application of deep LVM approaches omitting the modeling of the data-generating to infer latent embeddings. Unlike VAEs that target on data reconstruction, these models are designed to exclusively focus on the encoding process to extract underlying latent factors.

CEBRA is a recently developed tool following this non-generative framework, functioning as a deep LV generalization of nonlinear independent component analysis (ICA) that leverages contrastive learning for mapping high-dimensional neural recordings into latent embeddings. ICA is a DR technique which attempts to decompose multidimensional signals into a set of statistically independent components [57]. CEBRA is agnostic to the recording modality, which enables to recover consistent and identifiable latent dynamics representations from both electrophysiology and CaIm data. Fur-

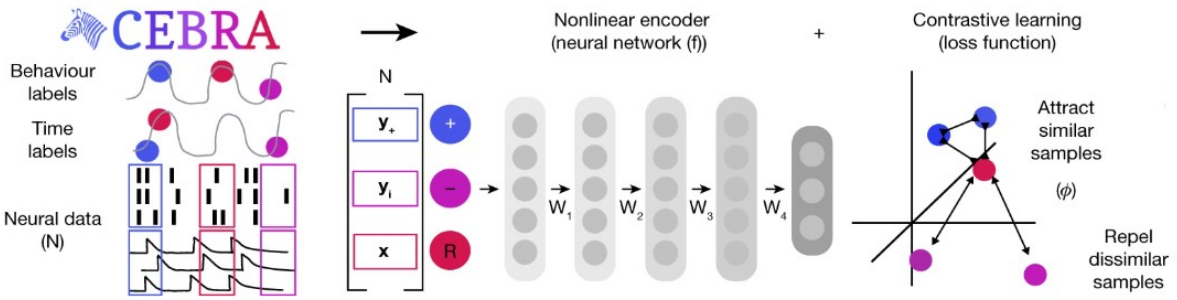

Figure 4: Contrastive learning data sampling scheme of CEBRA model. Adapted from [9].

thermore, it yields consistent embeddings across subjects and sessions, by effectively uncovering the population states related to the behavior. CEBRA offers unprecedented versatility, exploring both supervised and self-supervised frameworks over three modalities:

- **CEBRA-Time**: The model is trained under time contrastive objective in a self-supervised framework. It uniquely relies on temporal proximity to shape the latent space. Ultimately, this allows for the extraction of low-dimensional latent trajectories that reveal how neural population states evolve without requiring prior knowledge of the task. Because this mode excludes external behavioral variables, it imposes no restrictive assumptions on what factors influence the population, effectively preserving the intrinsic structure of the neural manifold. This makes it an ideal exploratory tool for uncovering a global latent space structure that exists independently of user-defined labels. Consequently, CEBRA-Time is commonly used as a first step for data visualization or as a baseline to validate subsequent hypothesis-driven analyses.

- **CEBRA-Behavior**: The model operates under a supervised framework by leveraging joint behavioral labels and neural data. These auxiliary variables encompass both continuous and discrete features of the subject, which serve as a reference for the model to discriminate embedding points. This mode conducts an hypothesis-driven analysis, where the user selectively chooses those variables that are hypothetically influencing neural activity. To verify whether the hypothesis is true, the resulting embedding should manifest alterations.

- **CEBRA-Hybrid**: This variant merges the contrastive frameworks of the previous modes, leveraging both time and behavior variables. It is mainly designed to disentangle the latent space to subspaces that are relevant to the given behavioral and residual temporal variance and noise. Using this mode allows to conduct hypothesis-driven analyses with temporal indices also influencing latent representations.

The effectiveness in decoding tasks of CEBRA models stems from its distinct methodology mainly focusing on extracting latent factors that directly correlate with specific

auxiliary variables. To achieve this, the framework utilizes contrastive learning, a technique that trains a non-linear encoder to discriminate between samples based on a similarity criterion, which is derived from temporal proximity and/or behavioral labels. By optimizing a classification-based objective the model distinguishes similar, positive pairs from dissimilar, negative pairs, ensuring a latent space that captures neural states correlating auxiliary variables patterns [58]. Consequently, CEBRA produces a coherent latent space that prioritizes behaviorally relevant features, facilitating high-performance downstream decoding across different sessions and recording modalities.

As illustrated in Figure 4, CEBRA draws samples from recordings, including both neural signal and behavioral labels. The model produces reference samples $x \in X^D$ and positive/negative samples $y \in Y^{D'}$ and clusters them into pairs (x, y). Positive samples should manifest similarity with respect to any of the auxiliary labels to distinguish them from negatives. The user defines how samples are drawn and paired to determine which neural population states should be considered as positive or negative pairs, according to the auxiliary label. Depending on the model's receptive field, each sample x or y typically represents a window of neural activity and/or context variables. Then, CEBRA model is trained for mapping neural activity from each sample to a lower-dimensional space Z^E by optimizing non-linear encoding neural networks $f : X \to Z$ and $f' : Y \to Z$ using a contrastive objective. While the feature extractors are typically identical ($f = f'$) for single-session experiments where reference and positive/negative samples are drawn from the same signal space, this notation reflects the framework's flexibility to integrate multi-session datasets with varying input dimensionalities or recording modalities. In particular, the input samples get paired and projected into the latent space in which they are evaluated under a similarity objective $\phi(z, z')$, where $z = f(x)$ and $z' = f'(x)$. This metric must provide higher scores to positive pairs, in order to attract similar samples, and lower scores to negatives to repel them from references. Therefore, CEBRA framework includes for similarity metric the normalized dot product $\phi(z, z') = z^T z'$ and negative mean squared error $\phi(z, z') = - \mid\mid z - z' \mid\mid^2$. The optimization objective relies on this similarity metric, which corresponds to $\psi(x, y)$ and is scaled by a temperature hyperparameter τ.

$$\psi(x, y) = \frac{\phi(f(x), f'(y))}{\tau} \tag{11}$$

The encoders are optimized to discriminate between similar and dissimilar samples through InfoNCE (Information Noise-Contrastive Estimation) loss function, which allows extract meaningful latent representations using a classification-based objective function. As shown in Eq. (12), InfoNCE minimization effectively guides the model to attract and repel positive and negative samples respectively through the similarity measure. Note that this loss function is an expectation, as x and y are drawn from their respective distributions: references ($x \sim p(x)$), positives ($y_+ \sim p(y \mid x)$) and negatives ($y_1, ...y_n \sim q(y \mid x)$). During each training iteration, CEBRA employs a specialized sampling scheme to construct contrastive pairs from a batch of size n. For every reference sample x in the batch, the model draws a single positive sample y_+ based on a user-defined criterion and multiple negative samples $\{y_1, ..., yn\}$ to serve as distractors. This sampling yields one positive pair and n negative pairs per reference,

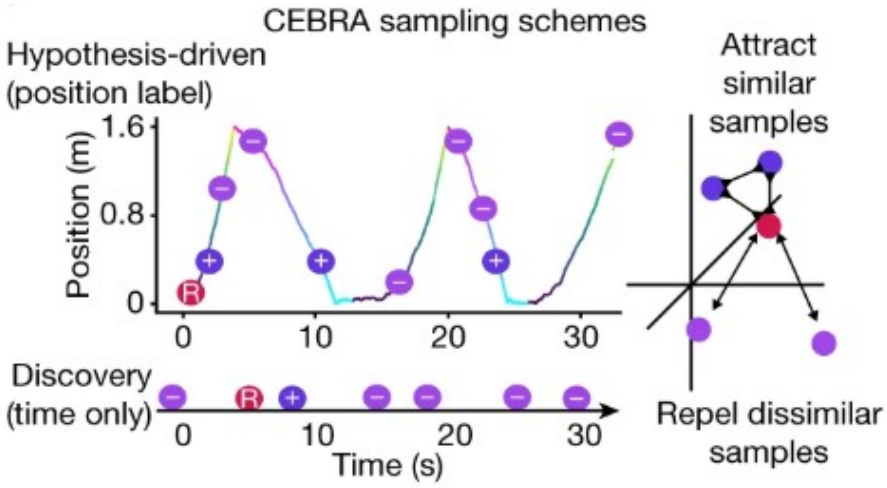

Figure 5: Sampling scheme of CEBRA for time only and position label models. Adapted from [9].

which are compared using the similarity measure to optimize the InfoNCE loss function. When hardware permits, CEBRA can utilize the full dataset as a single batch, allowing for a better approximation of the underlying positive and negative distributions and significantly improving the quality and consistency of the resulting latent embeddings.

$$\mathcal{L}_n[\psi] = \mathbb{E}[-\psi(x, y_+) + log \sum_{i=1}^{n} e^{\psi(x,y_i)}] \tag{12}$$

The CEBRA framework allows to configure the positive-pair distribution $p(y \mid x)$ and the negative-pair distribution $q(y \mid x)$ to selectively leverage auxiliary variables, thereby shaping the geometry of the resulting latent embeddings. Invariance to specific behavioral labels, or nuisance variables that do not provide any information, such as session ID, is achieved by enforcing that both positive and negative distributions are sampled uniformly across that factor. Because the invariant factor is distributed identically in both positive and negative pairs, it yields zero discriminative information for the InfoNCE objective. Consequently, the model learns that representing that factor becomes sub-optimal, and it is discarded from the embedding space.

In the simplest case, negative sampling follows the empirical distribution, which involves returning a randomly chosen index from the dataset. Alternatively, it can follow an imposed uniform distribution over a categorical variable to enforce invariance. In contrast, the procedure for positive sampling depends on the nature of the auxiliary variable. For discrete variable k_t, the algorithm filters the entire dataset for samples that share the same categorical value as the reference and selects one uniformly. For continuous variables, the user specifies a set of time offsets $\tau \in \Delta$ which define the temporal distance used to find related data points and informs the algorithm about the specific time-scale of interest for the analysis. In the case of time-only discovery-driven mode, the algorithm designs the distribution $p(y \mid x)$, depending on the tem-

poral distance set by the time offset from the reference samples drawn a from uniform prior $p(x)$. Hence, for a reference sample x at given instant t, the algorithm picks the sample defined by $t + \tau$. While for continuous auxiliary variables c_t in the hypothesis-driven mode, the user-specified time offsets Δ are used to construct an empirical distribution of behavioral dynamics. This distribution is defined as a set of differences $\mathcal{D} = \{t \in [0, T], \tau \in \Delta : c_{t+\tau} - c_t\}$, which captures the characteristic variations of the behavioral label over the chosen timescales. To identify a positive pair for a reference sample recorded at time t, the algorithm uniformly samples a displacement d from the set $\mathcal{D}$. It then searches the entire dataset to obtain the sample whose behavioral state is closest to the target state defined by the reference sample's context modified by that distance $c_t + d$. Furthermore, it is possible to combine a continuous variable c_t with a categorical variable k_t for mixed sampling, by ensuring that the set of close to reference samples share the same categorical status. This sampling framework is highlighted in Figure 5 and allows the model for both modalities to learn to discriminate positives over negatives, all guided by an auxiliary label.

Furthermore, this sampling scheme extends to multi-session and multi-animal experiments. This flexibility allows the framework to jointly leverage diverse data sources to discover animal-invariant features and produce more robust, consistent latent spaces. The critical condition for such joint training is that the dimensionality of the auxiliary variables must be shared across all sessions. Leveraging this capability, CEBRA successfully merged electrophysiology NP and CaIm data from the visual cortex of multiple mice to produce joint embeddings [9].

In contrast to generative frameworks, CEBRA exclusively targets data-encoding over the explicit reconstruction of raw neural signals [9]. While this specialization facilitates high-performance decoding, it requires to discard neural variability as noise, including trial-to-trial spiking fluctuations, which may lead to an inaccurate modeling of the neural dynamics [4]. Furthermore, these models may overlook potentially significant fine temporal structures or hidden contextual variables that are not captured by the user-defined auxiliary labels. Despite these limitations, CEBRA remains clearly as the SOTA of LVMs for latent representations, yielding embeddings with the best decoding capacity of all the available options [9].

However, contrastive learning can be integrated in generative frameworks, such as VAE, by implementing a contrastive term regularizing the objective function of the model, controlled by an hyperparameter. Therefore, the conventional generative approaches could leverage from behavioral labels to shape the neural manifold. This alternative is explored in further sections.

2.6 Market Analysis

The development of LVMs has followed a clear trajectory from fundamental scientific inquiry to translational and commercial applications. Early work before 2010s focused on uncovering the principles of neural computation, relying on relatively simple linear statistical models to understand how neural activity encodes stimuli and behavior [7], [59], [60]. These approaches provided important insights but were limited in their ability to capture complex, nonlinear neural dynamics. A few years later, initial commercial

and applied research efforts shifted the focus toward neural decoding, driven by early advances in machine learning and increased data availability. During this phase, models were designed to predict behavior or external variables from neural activity, enabling practical applications such as brain–computer interfaces (BCIs) and neuroprosthetics [61].

More recently, particularly since 2020, the field has entered a new phase characterized by deep LVMs specialized in representation learning techniques that capture more accurately the underlying states of neural populations during behavior, largely enabled by significant advances in neural recording technologies [8]. This paradigm emphasizes learning structured and informative latent representations of neural activity, with the goal of improving interpretability and providing deeper insights into the neural mechanisms underlying behavior [3]. Deep LVMs have demonstrated clear advantages, including improved generalization across tasks and subjects, enhanced decoding performance and scalability to large datasets [8].

Academic institutions remain the primary drivers of innovation in neural representation learning. Leading contributions come from interdisciplinary research groups combining neuroscience, ML and biomedical engineering. These entities are responsible for developing novel LVMs and typically publish both their methods and open-source implementations. For example, models such as CEBRA have been accompanied by patents, while still being disseminated through scientific publications [9]. In parallel, research institutions also play a key role in curating and sharing datasets through public repositories such as DANDI, facilitating reproducibility and large-scale evaluation [8]. The industrial landscape, in contrast, is dominated by neurotechnology and artificial intelligence companies, which focus primarily on data acquisition, hardware development, and clinical translation. Many of these companies invest in research and development of LVM approaches to improve the robustness and efficiency of real-time decoding systems for BCIs. Their efforts are particularly oriented toward deployable solutions in clinical and commercial environments [62].

In clinical settings, LVMs are increasingly being explored as key components in the treatment of neurological and neuro-degenerative disorders, particularly through their role in decoding neural signals within BCI systems [63]. Future trends suggest that the clinical adoption of BCIs could significantly enhance both the understanding and treatment of such conditions. These technologies rely heavily on accurate and generalizable decoding systems, where LVMs play a central role.

Looking ahead, continued progress in this field may lead to the emergence of foundation models for neural data, which correspond to large-scale LVMs capable of generating consistent and transferable representations across brain regions and tasks. Such models would represent a major step toward unified and scalable neural decoding frameworks [64].

3 Concept engineering

This section presents the engineering design and conceptual framework of the project, mainly dedicated to benchmarking LVMs through the systematic evaluation of their generated embeddings. The primary objective of the benchmark is to assess embedding quality with respect the decoding performance of behavioral labels and the preservation of global neural signal information. These measurements enable the quantification of the ability of LVM representations to encode behavioral context while preserving relevant neural dynamics. This design phase focuses on establishing a robust, reproducible and scalable evaluation framework capable of comparing multiple LVMs under consistent experimental conditions.

Several possible approaches for conducting the benchmark were explored and evaluated during the design process, considering the project objectives, the available computational resources, the reproducibility constraints and the expected extensibility of the framework. After comparing multiple alternatives, the selected solution was the implementation of a modular benchmarking pipeline, separating the models into three categories: baseline LVMs, deep non-generative LVMs (CEBRA) and deep generative models (GP-VAE). The proposed solution is to explore two datasets analyzing the activity of distinct brain regions regarding exclusively a positional label to evaluate the capability of the LVM to produce embeddings that accurately reconstruct the behavior and the neural signal.

Overall, the presented engineering solution aims to provide a rigorous and extensible framework for analyzing LVM embeddings, prioritizing methodological rigor, computational feasibility and experimental flexibility. This ensures that the benchmark remains both scientifically relevant and practically usable for future research and comparative studies.

3.1 Datasets

After analyzing multiple options, the proposed solution explores two experimental scenarios in which neural activity is coupled with an auxiliary contextual variable describing the continuous position of the subject. Consequently, the recorded neural dynamics are expected to contain encoded behavioral information related to spatial position, which the LVMs aim to capture within their learned representations. Within the scope of this study, positional information constitutes the most suitable behavioral label for evaluation because it provides an interpretable and continuous representation of behavior. This facilitates both the visualization and the understanding of the relationship between the learned embeddings and the underlying neural dynamics. By comparing embeddings with positional trajectories, it becomes possible to directly assess whether the latent representations preserve meaningful behavioral structure and whether spatial information is coherently organized within the latent space.

Alternative behavioral variables, including visual stimuli and more complex task-related behaviors, were also considered during the conception phase. However, these options were ultimately discarded due to their higher complexity and the difficulty of accurately characterizing the relationship between embeddings and behavior using the cur-

rent benchmarking framework. Such tasks would likely require more advanced analytical tools and significantly more computational resources to provide reliable and interpretable evaluations. Consequently, the use of positional information was considered the most appropriate compromise between interpretability, experimental rigor and feasibility.

The selected datasets were collected by the CEBRA package, provided from external sources (`https://github.com/AdaptiveMotorControlLab/CEBRA`). The reliability of these datasets for LVMs have previously been validated using the CEBRA model itself. This choice provides an important experimental reference point for the benchmark, as it enables the decoding performance of the implemented pipeline to be compared against established and previously reported results. Furthermore, both datasets provide sufficient number of neurons and temporal samples to support population-level analyses of latent representations, and their associated contextual variables also provide high-quality behavioral information describing the evolution of the subject's behavior over time.

Regarding preprocessing, the datasets were already provided with the necessary inherent preprocessing steps applied to the neural recordings. To ensure fairness and consistency across the benchmark, all evaluated models receive the same preprocessed data. Additional preprocessing operations were introduced only for the GP-VAE models, specifically to satisfy model's input requirements, while preserving the intrinsic temporal and structural properties of the neural recordings. This design decision ensures that performance differences arise primarily from the representational capabilities of the models themselves rather than from inconsistencies in data preparation procedures.

3.1.1 Rat Hippocampus Dataset

The rat hippocampus dataset is provided by Grosmark et al. [65]. It contains multicellular electrophysiology recordings of putative pyramidal neurons from the CA1 region of the hippocampus of four male Long-Evans rats: Achilles, Buddy, Gatsby, and Cicero, coupled to context labels. The recordings were obtained during navigation experiments in which each subject repeatedly traversed a 1.6-meter linear track while receiving water rewards at both ends of the track. Neural activity was simultaneously monitored from populations ranging between 48 and 120 neurons depending on the subject. Each session was segmented into trials corresponding to complete round trips along the track. In parallel, the position and running direction of the animal were recorded and associated with the neural activity as behavioral labels. In this case, a representative session contains more than 10,000 time-points with associated position and direction labels. Taking into account that even in the best-case scenario, the number of recorded neurons slightly exceeds 100 units, dataset's size can be considered as small. Following the preprocessing methodology proposed by Zhou et al., spike trains were discretized through spike binning using temporal windows of 25 millisecond [66].

An example of the experimental session can be observed in Figure 6. During the recording, the subject repeatedly performs navigation trials along the linear track, producing a periodic structure in both the position and direction labels. This repetitive

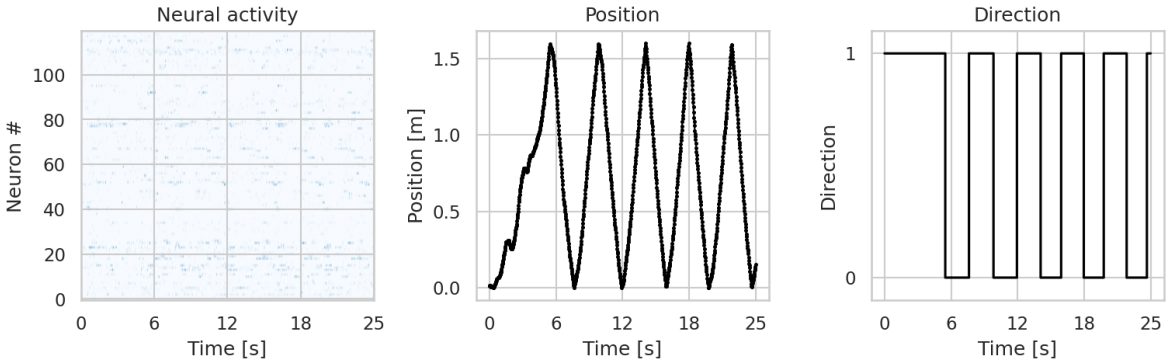

Figure 6: Example of neural activity and behavioral labels at the first 1000 time-steps extracted from the Achilles rat hippocampus dataset. Adapted from Grosmark et al. [65].

behavioral pattern is beneficial for latent representation learning, as the auxiliary labels evolve with limited variance across trials, facilitating the identification of consistent relationships between neural activity and behavior.

Within the conceived benchmark, only the recordings corresponding to the subject *Achilles* are considered for analysis. This selection was motivated by the fact that *Achilles* provides the largest recorded neural population among the four available subjects, with recordings comprising up to 120 neurons. A larger neuronal population enables more robust analyses of latent representations and improves the reliability of population-level embedding evaluations.

3.1.2 Macaque Reaching Dataset

The macaque dataset originates from the experimental recordings performed by Chowdhury et al. [67]. It consists of spiking recordings obtained from Area 2 of the primary somatosensory cortex (S1) during a single-session experiment conducted on a rhesus macaque subject. During the experiment, the animal performed center-out reaching tasks using a manipulandum, a device commonly used to measure motor skills, reaction time, or learning in neuroscience research, allowing the study of neural activity associated with limb movement and sensorimotor processing. The dataset includes both active and passive trials, in the same proportion, as described in [67]. The experimental trials encompassed a positional range of approximately -15 to 15 centimeters in both the x and y axes, where 288 neurons were recorded.

In the active trials, the monkey voluntarily executed center-out reaching movements toward one of eight spatial targets presented around a central position. In contrast, during passive trials, the monkey remained in a holding state while an unexpected external force of $2N$ was applied to the manipulandum, producing a passive displacement of the limb toward one of the same eight target directions. This experimental design enables the analysis of neural representations associated with both self-generated and externally induced movements. As illustrated in Figure 7, the trajectories of the manipulandum differ clearly between these two conditions. Active trials exhibit more controlled and goal-directed movements, resulting in smoother and more stereotyped

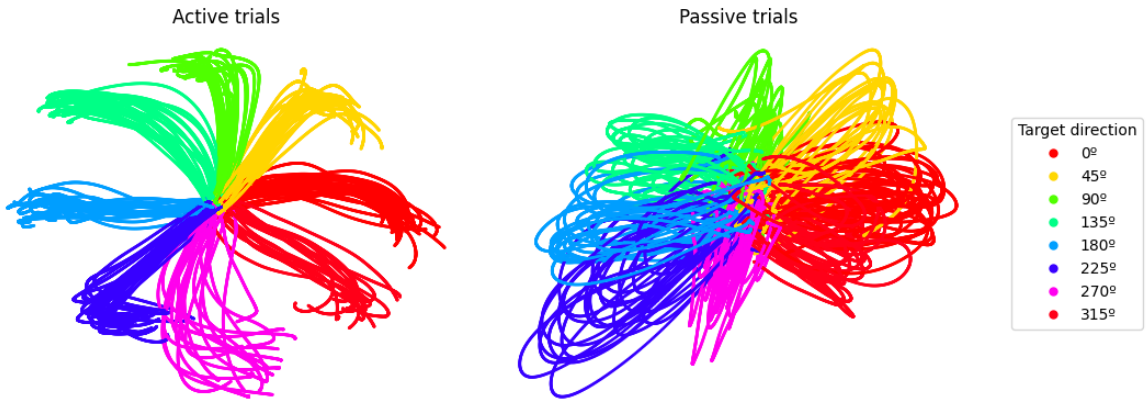

Figure 7: Hand position trajectories during experimental session of the monkey. Adapted from Chowdhury et al. [67].

trajectories toward the target directions. In contrast, passive trials produce more dispersed and irregular trajectories, reflecting the externally imposed perturbations and the absence of voluntary motor control. For the scope of this thesis, the analysis is restricted exclusively to the passive trials. Restricting the dataset to passive perturbations eliminates confounding variables associated with motor planning and task-dependent cognitive constraints inherent to learned voluntary behaviors. Consequently, this approach isolates the feedback loops of the proprioceptive system. This provides a clean paradigm to investigate how external stimuli modulate and reorganize neural manifolds, rather than voluntary decisions like in the previous dataset.

Following the preprocessing methodology proposed by Pei et al. [8], all trials were temporally aligned with respect to movement onset, considering temporal windows ranging from -100 milliseconds to 500 milliseconds. Neural spike trains were discretized through spike binning using 1 millisecond temporal windows and subsequently smoothed by convolution with a Gaussian kernel of 40 milliseconds standard deviation. As each trial consists of 600 samples, and the experimental session includes a 364 trials for both passive and active conditions, the dataset is larger than in the previous case, but it still remains computationally manageable and affordable. This preprocessing procedure reduces high-frequency variability while preserving the temporal structure relevant to movement-related neural dynamics.

3.2 Model Implementation and Training

An optimal implementation and training strategy should provide a standardized pipeline in which all models are trained under identical conditions, ensuring reproducibility, reliability and consistency across experiments. The training framework should require only minimal modifications, with model-specific preprocessing adjustments introduced only when strictly necessary.

Neural data is partitioned into training and test subsets following a conventional protocol. Models are fitted solely on the training data, where they learn latent representations from the intrinsic structure of the neural signals. In case of supervised variants, the

models have access to context variables besides the signal. It is crucial that these two signals are temporally coupled an evenly split to avoid any kind of leakage during learning. Once training is complete, the learned model is applied to the unseen test data to generate latent embeddings. During this stage, no model parameters are updated. Instead, the test data are projected into the latent space using the representations learned from the training set, ensuring a strict separation between the training and evaluation phases.

3.2.1 Baseline Models

The benchmark includes several techniques as baseline models, namely PCA, UMAP, and t-SNE. Since these approaches are primarily intended as baseline methods rather than optimized predictive systems, the training and hyperparameter optimization procedures are intentionally kept simple. In particular, for these models focuses exclusively on the latent dimensionality as the adjustable hyperparameter. This parameter directly determines the size of the embedding space and therefore has a significant impact on representation quality and decoding performance. Other model-specific hyperparameters are not exhaustively explored and instead remain fixed to standard or commonly adopted configurations. This design decision reduces computational overhead while still allowing meaningful comparisons between baseline and advanced models.

Due to the relative simplicity and reduced parameterization of PCA, UMAP and t-SNE, an additional validation stage was considered unnecessary within the scope of the benchmark.

3.2.2 Non-generative Deep LVM (CEBRA)

The CEBRA models are trained following the official PyTorch API guidelines and procedures provided in the online CEBRA tutorials and user documentation (`https://cebra.ai/docs/`). The primary objective is to reproduce, as closely as possible, the configurations and experimental settings presented in the original CEBRA studies for each dataset, while introducing only minimal adjustments required for integration into the benchmark framework.

Following the methodology recommended in the official tutorials, no explicit validation subset is defined during the training process for embedding generation. Within the context of CEBRA, the inclusion of a validation split is not strictly necessary due to the nature of the training protocol and the objectives of the benchmark. In most cases, models are trained without a validation split and when a validation subset is employed, it is generally not used for training regularization or hyperparameter optimization, but to qualitatively assess whether the generated embeddings remain coherent and generalizable or to perform additional shuffle tests aimed at verifying that the model effectively captures meaningful temporal correlations.

Rather than optimizing solely for predictive performance, the benchmark is designed to study the effect of different architectural and training hyperparameters on the structure and interpretability of the learned latent spaces. Consequently, hyperparameter exploration remains intentionally limited and controlled to a defined selection of values.

Another important consideration emphasized in the CEBRA guidelines is the prevention of the double dipping effect, where information from evaluation data unintentionally influences the training process [68]. To avoid this issue, all embeddings used for evaluation are generated from held-out data that remains completely unseen during training. This ensures that the resulting performance metrics accurately reflect the generalization capabilities of the learned representations.

3.2.3 Generative Deep LVM (GP-VAE)

The version of GP-VAE implemented in this project works with windowed inputs, therefore requiring temporal neural data to be preprocessed into sequential windows whose size is controlled and regularized through a dedicated hyperparameter. This sequential representation allows the model to capture temporal dependencies and smooth latent transitions across consecutive neural activity samples, which is particularly important for modeling neural dynamics.

The models are additionally regularized using an early-stopping mechanism designed to maximize the number of training iterations while preventing overfitting. GP-VAE requires the inclusion of a validation subset explicitly used to guide the training process and monitor convergence.

In addition, the presenting project explores a supervised version of GP-VAE, implemented by incorporating a contrastive term into the training objective, following a strategy conceptually similar to CEBRA.

$$\mathcal{L}_{\theta,\phi} = -\mathbb{E}_{q_\phi(z|x)}[\log(p_\theta(x \mid z)] + \beta D_{KL}(q_\phi(z \mid x) \mid\mid p_\theta(z)) + \lambda_c \cdot \mathcal{L}_{\text{contrast}} \tag{13}$$

where λ_c hyperparameter regularizes the contrastive term loss.

$$\mathcal{L}_{\text{contrast}} = \frac{1}{B^2} \sum_{i,j} \left(\tilde{D}^z_{ij} - \tilde{D}^b_{ij} \right)^2 \tag{14}$$

The contrastive loss works by minimizing the squared difference between two normalized pairwise distance matrices: one computed in the latent space $\tilde{D}^z_{ij} = \|z_i - z_j\|^2 / \bar{D}^z$ and one in the behavioral space $\tilde{D}^b_{ij} = \|b_i - b_j\|^2 / \bar{D}^b$. Here i, j index pairs of windows within a batch, not individual coordinates of z or b. B is the *batch size*, and B^2 normalizes by averaging over all pairs. The behavioral label $b_i \in \mathbb{R}^1$ is the rat's position at the last timestep of window i.

The loss enforces the geometry of the latent space to mirror the geometry of behavior space. Two windows with similar rat positions ($\tilde{D}^b_{ij}$ small) are pushed to have similar latent representations ($\tilde{D}^z_{ij}$ small), and vice versa. Normalization by the respective batch means $\bar{D}^z$ and $\bar{D}^b$ makes the two distance scales comparable before computing the discrepancy, preventing one from dominating due to scale differences.

This modification allows the application of contrastive learning principles within a generative LVM, combining the representational advantages of probabilistic generative

modeling with supervision guided by behavioral relationships. As a result, the supervised GP-VAE is expected to produce latent spaces that preserve both temporal neural structure and behaviorally relevant information. This generative model stems from the Master's Thesis of Berta Ros, and is currently under development in the Abante lab and has not yet been published (private repository within the lab).

3.3 Model Evaluation

The evaluated models are assessed through a set of reconstruction-based metrics designed to measure how much relevant information is preserved within the produced embeddings. Two different types of signals are reconstructed from the latent representations: the positional label and population activity signal derived from high-dimensional neural data.

To perform these reconstruction tasks, two different decoder are employed, both formulated as regression models: a linear regressor and a non-linear regressor. The objective of this dual-decoder strategy is to analyze whether the embeddings preserve information that can be recovered through simple linear relationships or whether more complex non-linear structures are required to decode the latent representations effectively. This distinction is particularly relevant when comparing different LVM, as some approaches may prioritize linearly separable representations while others may encode information in non-linear manifolds.

Only regression-based decoders are considered in the benchmark because evaluated variables are continuous in nature. Both datasets involve continuous positional trajectories, while the reconstructed neural population signal is also represented as a continuous quantity. Classification approaches were initially considered during the conception phase, but they were ultimately discarded because the categorical information present in the datasets, which mainly consists of direction labels, was implicitly encoded within the continuous trajectories themselves.

The decoding and reconstruction performance of the models are primarily evaluated using the coefficient of determination, denoted as R^2. This metric measures the proportion of variance in the target signal explained by the reconstructed output, thereby assessing the predictive capability of a model [69], as defined by the following expression:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{15}$$

where $\bar{y}$ denotes the mean value of the target signal, y_i the ground-truth target values, and $\hat{y}_i$ the corresponding predictions generated by the model. Higher R^2 values indicate stronger agreement between the reconstructed and original signals.

Unlike several other performance metrics, the R^2 score ranges from $-\infty$ to 1. Negative R^2 values indicate that the model performs worse than a naive predictor that always outputs the mean of the target data. Such results generally suggest that the selected model is inappropriate for the task, poorly fitted, or unable to capture the underlying

structure of the data [70].

3.3.1 Linear Regressor

The linear decoder employed in the benchmark is based on a standard linear regression model. This approach estimates a set of coefficients $w = (w_1, ..., w_p)$ that define a linear relationship between the embedding coordinates and the target signal [71]. The model is trained by minimizing the residual sum of squares between the observed targets in the dataset and the values predicted through the linear approximation. The predicted signal can be expressed as:

$$\hat{y} = w_1 z_1 + w_2 z_2 + ... + w_p z_p + b \tag{16}$$

where z_i represents the embedding vector, w_i the learned regression coefficients, and b the bias term.

The optimization objective of the model is to minimize the residual sum of squares (RSS):

$$\text{RSS} = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{17}$$

where y_i denotes the true target value and $\hat{y}_i$ the value reconstructed by the decoder.

The purpose of including this simple linear model within the benchmark is to provide a baseline against which a more expressive non-linear decoder can be compared. By evaluating reconstruction performance under purely linear transformations, it becomes possible to determine whether the embeddings organize information in a linearly accessible manner. Poor linear decoding performance may indicate that the relevant information is encoded through more complex non-linear structures within the latent space [46].

3.3.2 Non-linear Regressor

To achieve more accurate decoding of the latent representations, a non-linear regressor is also implemented within the benchmark framework. The objective of this decoder is to capture complex non-linear relationships that may exist between the embeddings and the reconstructed signals. While linear regression evaluates whether the latent space preserves information in a linearly accessible form, non-linear decoding allows the analysis of more intricate structures that may emerge in high-dimensional embedding manifolds.

The selected non-linear decoder is based on XGBoost, an optimized and distributed gradient boosting library designed to provide high computational efficiency, flexibility, and scalability. XGBoost implements supervised learning algorithms within the Gradient Boosting framework, where the final prediction is obtained through the additive combination of multiple weak learners, typically decision trees[72].

In gradient boosting, the model is built iteratively by sequentially adding new trees that attempt to correct the residual errors produced by the previous ensemble. At each iteration, the algorithm minimizes an objective function composed of a loss term and a regularization term:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{N} l(y_i, \hat{y}_i^{(t)}) + \sum_{k=1}^{t} \Omega(f_k) \tag{18}$$

where $l(y_i, \hat{y}_i^{(t)})$ represents the loss function measuring the discrepancy between the true target y_i and the prediction $\hat{y}_i^{(t)}$, while $\Omega(f_k)$ denotes the regularization term associated with the complexity of each decision tree f_k. The prediction generated by the model is obtained through the sum of the outputs of all trees:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i) \tag{19}$$

where f_k corresponds to an individual regression tree and x_i represents the embedding vector associated with sample i.

To ensure consistency and fairness across experiments, the hyperparameters of the XGBoost decoder are kept fixed for all evaluated model groups and datasets. By maintaining identical decoder configurations, differences in reconstruction performance can be attributed primarily to the quality of the embeddings themselves rather than to decoder-specific optimization procedures.

3.3.3 Positional Decoding Analysis

The objective of this decoding analysis is to reconstruct positional information from the embeddings. This evaluation aims to measure how much information related to position is encoded within the latent representations learned from neural activity.

All positional labels employed in this project describe continuous coordinates associated with the experiment $c(t)$. In the rat hippocampus dataset, these coordinates correspond to the spatial position of the subject in a single axis along the navigation task (1D-vector), while in the macaque dataset they describe the position of the subject's hand during the reaching movements on a squared XY plane (2D-vector).

As previously mentioned, decoding performance is quantitatively evaluated using the R^2 score obtained from the predictions generated by decoders trained on the embeddings produced by the LVMs. In addition to quantitative evaluation, the embeddings are also analyzed qualitatively through dedicated visualization techniques. Specific embedding plots are generated to verify whether the latent representations exhibit manifolds consistent with the underlying behavioral trajectories. These visual plots provide complementary evidence regarding whether the embeddings preserve meaningful positional organization beyond the numerical decoding metrics alone.

3.3.4 Global Activity Decoding Analysis

In many cases, latent embeddings may lose relevant information about the original neural signal from which they are derived. As a consequence, a model can achieve strong behavioral decoding performance while still failing to preserve meaningful properties of the underlying neural population dynamics. To address this issue, this benchmark incorporates an additional evaluation focused on measuring the correlation between the embeddings and the global neural activity.

Direct reconstruction of the raw neural signal from the embeddings is not considered reliable due to the high dimensionality and complexity of the neural recordings. Instead, the ground-truth reference signal $g(t)$ is defined as a derived representation of the neural population activity. Specifically, the neural signal is summarized by computing the mean of the activity across all N recorded neurons at each time point t over the duration of the session T, resulting in a one-dimensional signal that captures the global population dynamics over time.

$$g(t) = \begin{pmatrix} \frac{\sum_i^N x_i(t_1)}{N} \\ \frac{\sum_i^N x_i(t_2)}{N} \\ \vdots \\ \frac{\sum_i^N x_i(T)}{N} \end{pmatrix} \tag{20}$$

Consequently, the decoders are trained using the a global derive from the same neural signal that they already used for training. The resulting reconstruction metrics therefore measure the extent to which the latent representations maintain information correlated with the overall neural population activity.

4 Detailed engineering

In this section, the implementation of the selected solution is detailed. This includes all the detailed procedure regarding the training and evaluation of the LVMs. All generated code is available in a GitHub repository (URL:`https://github.com/AbanteLab/neural_latents_bench`)

4.1 Software Architecture

The implemented pipeline is organized around a sequence of independent yet interconnected modules responsible for data preparation, embedding extraction, decoding evaluation and global signal reconstruction.

These shared functionalities are centralized within a common `utils.py` module, which provides standardized methods used across all benchmark configurations. This design promotes code reuse and ensures consistency in preprocessing, evaluation and data handling procedures. Each benchmark dataset directory contains a folder containing the experimental datasets of different sessions or conditions, and a dedicated folder for every LVM evaluated in the framework.

Inside each model folder, two main files are included: a Python script responsible for training and evaluation and a Bash script used to launch experiments under different hyperparameter configurations. Each Python file generates both a trained model and the corresponding evaluation results associated with a specific experimental configuration. These files are compiled by `nlb` conda environment, including all the required packages to conduct the project. Conversely, each Bash file contains the execution commands required to launch the corresponding Python scripts under different combinations of hyperparameters. These configurations are handled through argument parsers, allowing all experimental settings to be explicitly defined at runtime without modifying the underlying source code.

To facilitate traceability and reproducibility, both models and result files include the relevant hyperparameters of the corresponding LVM within their identifiers. Trained models are stored separately in a dedicated models directory using the `.pt` format provided by PyTorch. In parallel, evaluation results are saved as `.csv` files inside the corresponding LVM directory, simplifying metric analysis and comparison across experiments.

Additionally, stored models include a label for identifying the category of the LVM within the models folder. In the case of results, grouping all outputs together was not considered practical, since although the evaluated models share common metrics and certain parameters, each LVM category also introduces its own specific hyperparameters and configuration variables. Keeping results separated by model type therefore improves readability and avoids unnecessary complexity in experiment management. The use of a unified embedding interface allows LVM to be evaluated under the same protocol, while standardized preprocessing ensures consistency across experiments.

The training workflow is implemented under a conditional structure that first searches

within the models directory for previously trained models. If a model with the same configuration already exists, the training stage is skipped and the stored model is directly loaded. This strategy avoids unnecessary retraining and prevents duplicating computationally expensive model checkpoints, significantly reducing execution time and storage redundancy. In contrast, pre-existing evaluation results are not restricted in the same way. Since `.csv` result files are considerably lighter than trained model checkpoints, new evaluation outputs can be generated and stored without introducing significant storage overhead. This allows experiments to be rerun and metrics to be updated more flexibly when needed.

To facilitate the analysis and visualization of the benchmark outputs, an additional `plot_results.py` module is developed for each dataset directory. This module loads selected trained models and their associated evaluation results from the benchmark folders and generates comparative visualizations across the different categories of LVM. The visualization framework enables the comparison of metrics of interest across multiple experimental configurations. The main objective of this module is to simplify the selection and inspection of relevant models, particularly those providing the best performance under specific criteria. Once selected, the framework allows visualization of the produced embeddings, reconstructed signals, and shared evaluation metrics. This supports both quantitative comparison and qualitative interpretation of the learned latent representation

For multi-parameter are executed within persistent `tmux` sessions on a linux machine, enabling long-running simulations to be managed efficiently and robustly, even in remote or distributed computational environments.

4.2 Benchmark Workflow

To conduct a meaningful benchmark, it is necessary to define the evaluation metrics used to compare the different models under study. In this work, the goal is to keep the comparison as close as possible to the intrinsic properties of the models by focusing exclusively on parameters that are directly comparable across architectures and that actively contribute to the formation of the learned representation manifold. According to this benchmarking criteria, only a limited set of hyperparameters is considered for the analysis, while those related to the optimization procedure, such as the learning rate, batch size and number of epochs or iterations, are not included as part of the benchmark analysis due to resources limitations. The set of considered hyperparameters for this benchmark analysis is collected in Table 1.

Once the Python implementations have been verified to run correctly, without bugs or issues related to training or data storage, and all project files have been properly organized, the hyperparameter evaluation phase is conducted for each benchmark category. To ensure robustness and reproducibility, all experiments are repeated across five different random seeds, controlled via a `seed` hyperparameter. In case of baseline models

For all models, the latent dimensionality was swept over the following set of values: (3, 5, 7) to ensure reproducibility. Higher dimensionalities are beyond the scope of this

Hyperparameter	Models	Description
`seed`	All	Random seed used to evaluate reproducibility and robustness across independent runs.
`latent_dim`	All	Dimensionality of the latent representation learned by the model.
`time_offset`	CEBRA	Temporal offset used to define positive sample pairs for contrastive learning.
`conditional`	CEBRA	Determines whether auxiliary behavioral information is incorporated during training.
`beta`	GP-VAE	Weight assigned to the latent regularization term in the loss function.
`lambda_contrastive`	GP-VAE	Weight assigned to the contrastive term in the loss function.

Table 1: Hyperparameters included in the benchmark analysis. Only hyperparameters that directly influence the learned latent representations are considered.

benchmark, which focuses more on low-dimensions to asses the capacity of the models to compress information.

4.2.1 Baseline Hyperparameters

For the baseline LVMs, only the `latent_dim` hyperparameter is evaluated, as it is the only parameter shared across all models in this category. Furthermore, exploring additional model-specific hyperparameters is beyond the scope of the baseline comparison, whose primary objective is to establish a reference performance against which more complex LVMs can be compared. Therefore, only the effect of latent dimensionality on the quality of the learned neural representations is investigated, as the goal of this project is not to optimize the baseline models but rather to provide a fair and consistent benchmark.

4.2.2 CEBRA Hyperparameters

Within the CEBRA framework, several hyperparameters allow for extensive model optimization, though only a specific subset is explored within the scope of this study due to their relevance in the context of this benchmark. The `model_architecture` parameter defines the encoder structure and determines the receptive field, corresponding to the number of time points per sample. Unlike the tested GP-VAE relying on recurrent architectures to process temporal relations, CEBRA primarily uses convolutional networks to process time-varying signals for extracting latent components, so it

captures temporal structure through its receptive field and contrastive sampling rather than through explicit signal windowing. However, the receptive field is not a standalone numerical parameter that can be adjusted directly; instead, it is implicitly defined and regularized by the choice of `model_architecture`. CEBRA offers only three pre-defined models with different receptive field, substantially limiting the exploration of this hyperparameter. Therefore, `model_architecture` get excluded from the benchmark and maintained fixed to recommended architecture using 10 time points per sample.

Beyond the latent dimensionality, the primary hyperparameters explored in this study were the time offsets and the conditional mode. The `time_offsets` parameter is a fundamental component of the CEBRA framework, as it directly regularizes the sampling process by defining the temporal distance between positive pairs. This parameter informs the algorithm about the specific time-scale of interest within the neural or behavioral data. In the context of this study, varying the time offsets provides a robust method to visualize and compare how different temporal dependencies influence the structure of the resulting latent space, with higher offsets typically increasing the difficulty of the learning task to potentially yield higher-quality representations. The `conditional` parameter allows for the selection of the contrastive paradigm, determining whether the model operates in a supervised or unsupervised capacity. To ensure a realistic range for `time_offsets` based on user's guide, the values (10, 20, 30, 40, 50) were selected.

Following the framework guidelines, the `batch_size` was set to the largest value supported by the device (2,048 samples). This choice maximizes the number of positive and negative sample pairs within each training iteration, leading to a more representative approximation of the underlying data distribution for contrastive learning. Consequently, it improves the stability and effectiveness of the contrastive objective during training. Furthermore, the models utilized the default InfoNCE criterion and cosine similarity metric, both of which are specifically tailored to the contrastive optimization objective. Future research may explore alternative metrics and distance functions to more rigorously characterize CEBRA's latent space performance.

4.2.3 GP-VAE Hyperparameters

Due to its generative nature, the GP-VAE includes several hyperparameters that directly influence the latent representations and the underlying generative process used to explain the observed neural activity. Consequently, unlike the baseline models, a broader set of hyperparameters is considered in the benchmark, as these parameters actively contribute to the structure and quality of the learned embeddings.

The GP-VAE implementation used in this project optimizes a loss function composed of three terms: a reconstruction term, a regularization term, and a contrastive term. Note that in this case, the GP-VAE is always supervised due to the presence of the contrastive term. The relative contribution of the regularization and contrastive components is controlled by the hyperparameters `beta` and `lambda_contrastive`, respectively. The objective of the benchmark is to evaluate how different combinations of these contributions affect the quality of the learned generative model, particularly in terms of its ability to reconstruct neural activity and preserve behaviorally relevant information

encoded in the latent representations. In addition to the latent dimensionality, the GP-VAE benchmark therefore investigates the influence of these two hyperparameters by evaluating both low and high values, allowing the analysis of the trade-off between reconstruction, latent regularization, and contrastive structure. For both regularization hyperparameters, the following values were explored: (0.1, 1.0, 100.0, 1000.0), in order to assess the effect of weak versus strong regularization on model performance.

Another relevant hyperparameter is the temporal window size, which determines the temporal context available to the model when learning relationships within neural activity. In this work, the window size is fixed to 50 time points, as empirical observations indicated that this value captures sufficient temporal information while maintaining a reasonable computational cost. Larger window sizes substantially increase memory requirements and training time, whereas smaller windows fail to adequately capture the temporal dependencies present in the neural recordings.

The batch size is determined through a dedicated helper function designed to maximize the utilization of the available data. Specifically, the function selects a batch size that is compatible with the chosen window size while minimizing the number of residual time points that cannot be assigned to complete windows, thereby reducing data wastage during training.

Other GP-VAE hyperparameters, such as those controlling the initialization of the kernel, are excluded from the benchmark. These parameters primarily affect optimization and initialization aspects of the model rather than the properties of the learned latent manifold and therefore fall outside the scope of the comparative analysis.

4.2.4 Evaluation Framework

To ensure a fair comparison across LVMs, all decoders are trained using the same configuration and default hyperparameters. Unlike linear regression models which are employed as a simple and interpretable baseline decoder without including any hyperparameter, XGBoost regressor models exhibit more complexity, with a specific set of adjustable hyperparameter, to be able to capture nonlinear relationships between latent representations and target variables. The hyperparameters of the decoders are kept fixed across all experiments, as optimizing decoder-specific parameters for each LVMs falls outside the scope of this benchmark.

The performance obtained under both decoding tasks is used to identify the most effective hyperparameter configurations for each model. The resulting scores are summarized in comparative tables, and the coefficient of determination R^2 achieved for positional decoding and global activity reconstruction is jointly analyzed to assess the versatility of the learned representations. The optimal values of these metrics for each model under the available dimensions are shown in barplots. In addition, to observe the tendency over the hyperparameter configurations, this evaluation framework includes violin plots. Furthermore, the embeddings corresponding to the best-performing configurations, based on highest R^2 score for each evaluation task, are visualized to qualitatively inspect the structure of the learned latent manifolds.

To facilitate the comparison of models across multiple objectives, a Pareto-front anal-

ysis is conducted for the R^2 score of both evaluation tasks. In multi-objective optimization, the Pareto front represents the set of solutions for which no objective can be improved without simultaneously degrading at least one other objective. Consequently, each point on the Pareto front corresponds to a non-dominated solution, while every solution outside the front is dominated by at least one Pareto-optimal solution [73]. In this work, Pareto-front analysis provides a framework for identifying hyperparameter configurations that achieve the best trade-off between behavioral decoding performance and neural activity reconstruction. Specifically, the Euclidean distance to the optimal point (1,1) is computed for each model, and therefore, the model with the smallest distance is selected as the best model. However, this Pareto analysis also considers the dimensionality of the latent variable models (LVMs) and evaluates how closely lower-dimensional versions remain to the Pareto frontier, thereby assessing the models' ability to retain performance while using fewer latent factors.

5 Results

In this section, results of the benchmark are presented and discussed. Following the defined structure of the project, the rat hippocampus dataset is analyzed first, followed by the macaque reaching dataset.

Model evaluation starts by reporting and comparing the optimal scores obtained by each LVM, considering how they behave over the number of latent factors. The dimensionality of the optimal embeddings provides meaningful insights about model's ability to compact information. For further application of the models, yielding latent representations that perform well with smaller dimensionalities is a clear benefit among other models which require way more latent factors to achieve similar performance. Consequently, for each dataset, the optimal R^2 scores in function of the latent dimensions are presented in both positional decoding and global signal preservation cases, exclusively for non-linear decoder (linear decoder graphics are in Appendix C). Non-linear decoder was selected over its linear counterpart because it achieved higher scores, in general. This involves that predicted labels can be described better with non-linear decoders rather than simpler linear models. For complementary visualization of the evaluation scores in deep LVMs, influenced by hyperparameter combinations, violin plots over latent dimensions of the R^2 values of non-linear decoder are presented in Appendix B

The learned embeddings are qualitatively analyzed to justify the observed performance differences and to identify the main features that characterize each representation, which provides insights of the manifold shaping strategies imposed by each LVM variety. All the embedding plots are collected in Appendix A. The effect of hyperparameters on deep LVMs' performance are discussed to determine their influence on encoding the neural manifold.

The optimal scores for these two objectives are plotted, comparing the joint performance across models, with a Pareto front constructed to highlight the best trade-offs. The optimal model selection is based on the euclidean distance to an optimal point $(1, 1)$ for the plotted metrics, which goes beyond the Pareto frontier in both cases. For each dataset, a model ranking is performed according to their distance relative to the optimal point. The resulting performance considering the dimensionality of the models is discussed.

5.1 Rat Hippocampus Benchmark

In this dataset, the recorded neural activity originates from hippocampus brain regions strongly associated with spatial navigation [65]. Consequently, the latent features extracted by the evaluated models are expected to reflect spatial representations analogous to internal cognitive maps. Under this interpretation, the learned embeddings can be viewed as low-dimensional representations of the spatial mapping encoded by brain's activity. Therefore, an effective latent representation should preserve the underlying structure of the navigational behavior, organizing neural activity according to the spatial trajectory followed by the subject.

In general, baseline LVMs show better performance in preserving the global neural sig-

nal, but they exhibit limited capacity for decoding position. In contrast, deep LVMs consistently achieve higher positional decoding performance, although they poorly preserve information about the original neural activity. This tendency can be clearly observed in the barplots (Figure 8 for positional decoding and Figure 9 for global reconstruction).

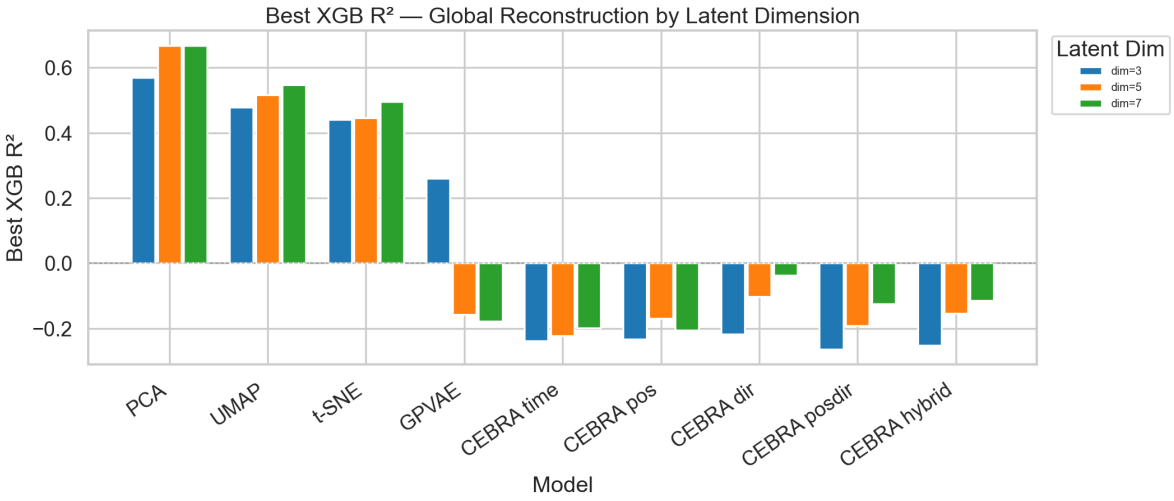

Figure 8: Barplot of the best R^2 scores for position decoding of each model variety using non-linear decoders over latent dimensions for rat hippocampus dataset

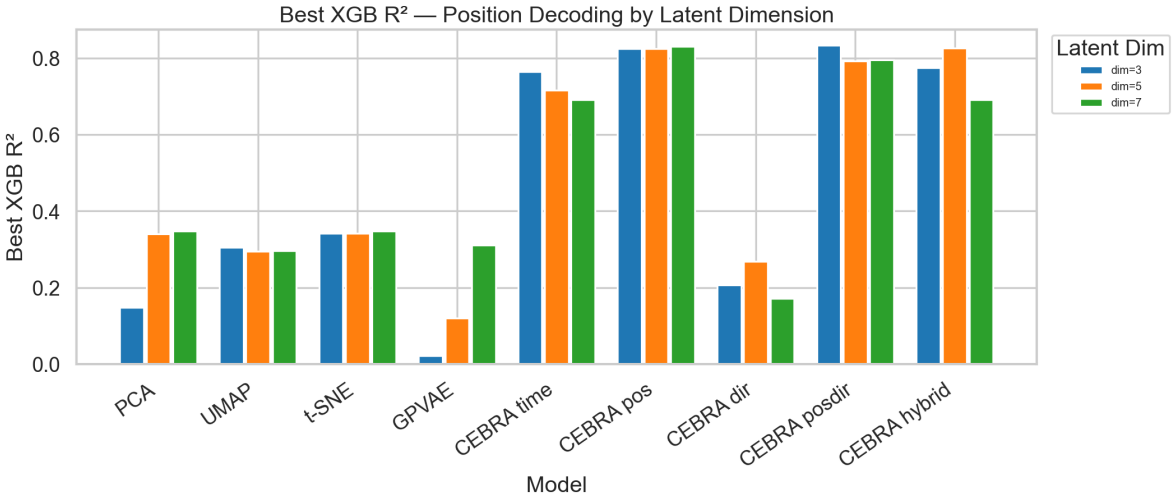

Figure 9: Barplot of the best R^2 scores for global activity reconstruction of each model variety using XGBoost estimators over latent dimensions for rat hippocampus dataset.

5.1.1 Baseline Results

The results obtained for the baseline LVMs indicate that these methods are not optimal for capturing the positional information encoded in the neural activity. Since they operate in a fully unsupervised manner, the embeddings are learned exclusively from the structure of the neural signal, without access to any behavioral labels. Neural activity is inherently variable, and the variance captured by these methods is not necessarily

associated with position. As a result, the learned embeddings often fail to organize the samples according to the positional label. However, because the embeddings are derived solely from the neural activity, baseline LVMs tend to preserve more information from the original signal than supervised or self-supervised approaches.

The compression capability of these methods vary depending on the model and the decoding task of the non-linear regressor. For instance, UMAP and t-SNE exhibit less accentuated performance drop when dimensionality diminishes, while PCA is more susceptible to smaller dimensionalities, especially in decoding position. This suggest that PCA models are not able to compress the data as well as the other baselines and require larger number of components to achieve comparable performance. However, when compared at highest dimensionality, PCA surpass significantly both models. So for this dataset, UMAP and t-SNE may compress better the data but achieve lower potential peak performance in comparison to PCA.

The embedding visualizations for baseline LVMs support this interpretation (see Figure A.1). Among the baseline methods, PCA exhibits the clearest positional encoding, with samples corresponding to different positions projected relatively far apart in the latent space. This separation becomes more evident when the embeddings are colored according to position. In contrast, the embeddings produced by UMAP and t-SNE show a stronger mixing of samples from different positions, without a clearly identifiable manifold structure associated with the positional label. The decoding results are consistent with these observations, as PCA achieves slightly better positional decoding performance than the other baseline methods.

Model	**Optimal Positional Decoding**		**Optimal Global Reconstruction**	
	R^2	**Latent Dim.**	R^2	**Latent Dim.**
PCA	0.3479 (xgb)	7	0.6675 (xgb)	7
t-SNE	0.3476 (xgb)	7	0.4953 (xgb)	7
UMAP	0.3047 (xgb)	3	0.5476 (xgb)	7

Table 2: Comparison of baseline LVM methods optimal performance in rat hippocampus dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin).

These results can be explained by the nature of unsupervised representation learning. Since the latent space is not conditioned by external labels, the models are unable to explicitly shape the manifold according to behavioral variables. Consequently, positional information can only emerge indirectly through patterns present in the neural activity. In practice, the variability of the neural signal and the complexity of the underlying population dynamics often prevent these methods from extracting representations that accurately encode position. Furthermore, the baseline methods rely primarily on the variance structure of the data, with PCA being restricted to linear transformations. This limitation may oversimplify the complex and non-linear dynamics present in neural population activity.

The quantitative results summarized in Table 2 show that PCA consistently outper-

forms the other baseline LVMs in both positional decoding and global activity reconstruction. Nevertheless, despite being the strongest baseline, its positional decoding performance remains substantially lower than that achieved by the deep LVMs.

5.1.2 CEBRA Results

The CEBRA models achieve the highest positional decoding performance among all the evaluated LVMs, but they also exhibit the poorest preservation of the global activity signal. This behavior reflects the contrastive learning paradigm employed by CEBRA, which explicitly shapes the neural manifold according to the external signal used during training. As a consequence, the latent representations become highly optimized for the behavioral variable, often at the expense of preserving information about the original neural activity.

The embedding visualizations support this interpretation (see Figure A.2). CEBRA variants trained with positional information recover the spatial structure of the task remarkably well, producing embeddings that resemble a map of the rat's trajectory. Although CEBRA-Time is trained without explicit behavioral labels, it still recovers part of this structure, demonstrating the ability of the contrastive framework to exploit temporal relationships within the neural activity. In contrast, the embeddings produced by CEBRA-Dir appear as a largely unstructured cloud of points, showing little evidence of a meaningful positional organization.

The poor performance of CEBRA-Dir can be explained by the nature of the supervisory signal. Direction is represented by a binary variable, which provides substantially less information for shaping the latent manifold than the continuous positional label. Consequently, the model is unable to learn rich associations between neural activity patterns and behavioral context. While the direction label introduces some supervision, it is insufficient to organize the latent space according to the underlying navigation dynamics. As a result, CEBRA-Dir behaves similarly to a weakly supervised or almost unsupervised LVM, while simultaneously sacrificing part of the signal preservation observed in the baseline models. For this dataset, it therefore represents the least effective CEBRA variant.

Model	**Optimal Positional Decoding**			**Optimal Global Reconstruction**		
	R^2	**Latent Dim.**	**Time Offset**	R^2	**Latent Dim.**	**Time Offset**
CEBRA time	0.7643 (lin)	3,	10	-0.0504 (lin)	7	10
CEBRA pos	0.8298 (xgb)	7	10	-0.0552 (lin)	5	10
CEBRA dir	0.2691 (xgb)	5	50	0.0838 (lin)	7	100
CEBRA posdir	0.8332 (xgb)	3	10	-0.0684 (lin)	7	10
CEBRA hybrid	0.8258 (xgb)	5	10	0.0643 (lin)	7	10

Table 3: Comparison of CEBRA methods optimal performance in rat hippocampus dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin).

The optimal performance of CEBRA models using non-linear decoders for both evalua-

tion tasks are collected on Table 3. As it was expected, CEBRA models are significantly superior to the rest of models when decoding position, with all supervised variants getting scores above 0.8 (except CEBRA-Dir). However, for global activity decoding, CEBRA performance was the weakest by far, with all the models showing negative values except CEBRA-Dir. This supports the hypothesis that auxiliary labels distort the preservation of the neural activity original signal from the embeddings.

The violin plots of CEBRA models for this dataset are available (see Appendix B) Among the CEBRA variants, position-supervised models consistently achieve the best and most stable position decoding (around 0.75–0.80) with minimal variance across dimensions 3, 5, and 7. However, this comes at the cost of global reconstruction, which is systematically negative for all CEBRA variants. CEBRA-Dir is the weakest variant, producing near-zero position decoding since direction labels carry little information about position in the hippocampal task. CEBRA-Time and CEBRA-Hybrid show higher variance across hyperparameter configurations, with wide bimodal distributions, suggesting these objectives are less stable at low dimensionality. Overall, latent dimensionality has little effect on median performance in the 3–7 range.

The results reported in Table 3 also provide insight into the effect of the hyperparameters. For positional decoding, the best performance is consistently obtained for a `time_offset` value of approximately 10, suggesting that this temporal scale captures the most informative neural relationships for the navigation task. For global activity preservation, however, larger `time_offset` values tend to produce better results in models trained with positional labels. A possible explanation is that larger temporal offsets reduce the strength of the contrastive constraints imposed on nearby samples, resulting in a less aggressive shaping of the latent space and therefore a smaller loss of information from the original neural signal.

For CEBRA models, latent dimensionality has a relatively minor effect on positional decoding performance. Similar decoding scores are obtained across a broad range of latent dimensions, especially in CEBRA-Pos,indicating that CEBRA is capable of capturing the relevant positional information even in low-dimensional latent spaces. Even some models, such as CEBRA-Hybrid or CEBRA-Time, perform better at lower dimensions. This observation also explains why the optimal latent dimensionalities reported in Table 3 differ considerably between models despite producing comparable decoding performance. In contrast, global activity reconstruction exhibits a clearer dependence on latent dimensionality, with larger latent spaces generally leading to better signal preservation. This result is consistent with the expectation that higher-dimensional embeddings can retain a greater amount of information from the original neural activity. However, despite this gradual improvement over dimensions, the result is still very poor. Therefore, for this dataset, CEBRA surpass by far other LVMs in positional decoding performance and compressing, but struggles to preserve information about the original data.

5.1.3 GP-VAE Results

For this dataset, GP-VAE model shows weak performance in positional decoding compared to CEBRA, and significantly lower performance in global activity reconstruction.

The supervised behavior of GP-VAE model allow it to surpass baseline, however it requires more latent dimensions to achieve decent performance, suggesting that it has poor compressing performance.

The best positional decoding performance is obtained with an R^2 score of 0.31 using a linear decoder, corresponding to hyperparameter values `lambda_contrastive` = 100.0 and `beta` = 0.1. In contrast, the best global reconstruction performance reaches an R^2 score of 0.26 using a non-linear decoder, obtained with `lambda_contrastive` = 0.1 and `beta` = 0.1. These results highlight a clear trade-off between optimizing for behavioral decoding and preserving global neural activity. As shown in the barplots (Figure 9 and Figure 8), the compressive capacity of GP-VAE show different trends depending on the evaluation metric. For lower dimensions, the model exhibits poor decoding of positions but preserves best the global activity, while for higher dimensions is the opposite. In fact, when scales to 5 latent factors to describe the embedding space, the global preservation scores begin to be negative. Consequently, increasing the dimensionality in GP-VAE models leads to better description of the positional label with the cost of losings information.

As the hyperparameter `beta` controls the strength of the regularization term in the ELBO, which constrains how closely the encoder posterior matches the prior distribution, the lower values associated to optimal performance for GP-VAE model suggest that the latent space to deviate from the imposed prior gaussian structure. Conversely, `lambda_contrastive` controls the weight of the contrastive term in the loss function. As expected, higher values of `lambda_contrastive` lead to improved positional decoding performance, since this term explicitly encourages the latent space to organize according to the behavioral structure. In contrast, lower values of this parameter result in better global activity reconstruction, as the latent space is less constrained by behavioral alignment and therefore preserves more information about the original neural signal.

The embedding visualizations are consistent with these findings (see Figure A.3). The latent space exhibits an approximately Gaussian structure, as expected from the variational formulation of the model. However, the positional organization is less clearly defined compared to CEBRA-based embeddings.

GP-VAE consistently struggles to encode positional information across all tested latent dimensionalities. This is likely because its Gaussian prior do not directly align the latent space with behavioral covariates, unlike CEBRA, which uses supervised contrastive learning to explicitly attract embeddings sharing the same label. However, it compresses better the original signal with lower dimensions, suggesting that this global activity signal can be captured by a Gaussian latent prior at minimal dimensionality.

5.1.4 Pareto Analysis

In Figure 10 it is shown the Pareto front obtained for this benchmark. As summarized in Table 4, the Pareto-optimal methods are PCA and CEBRA-Hybrid, since neither method can improve one objective without sacrificing the other.

Among all evaluated approaches, PCA achieves the highest global reconstruction score

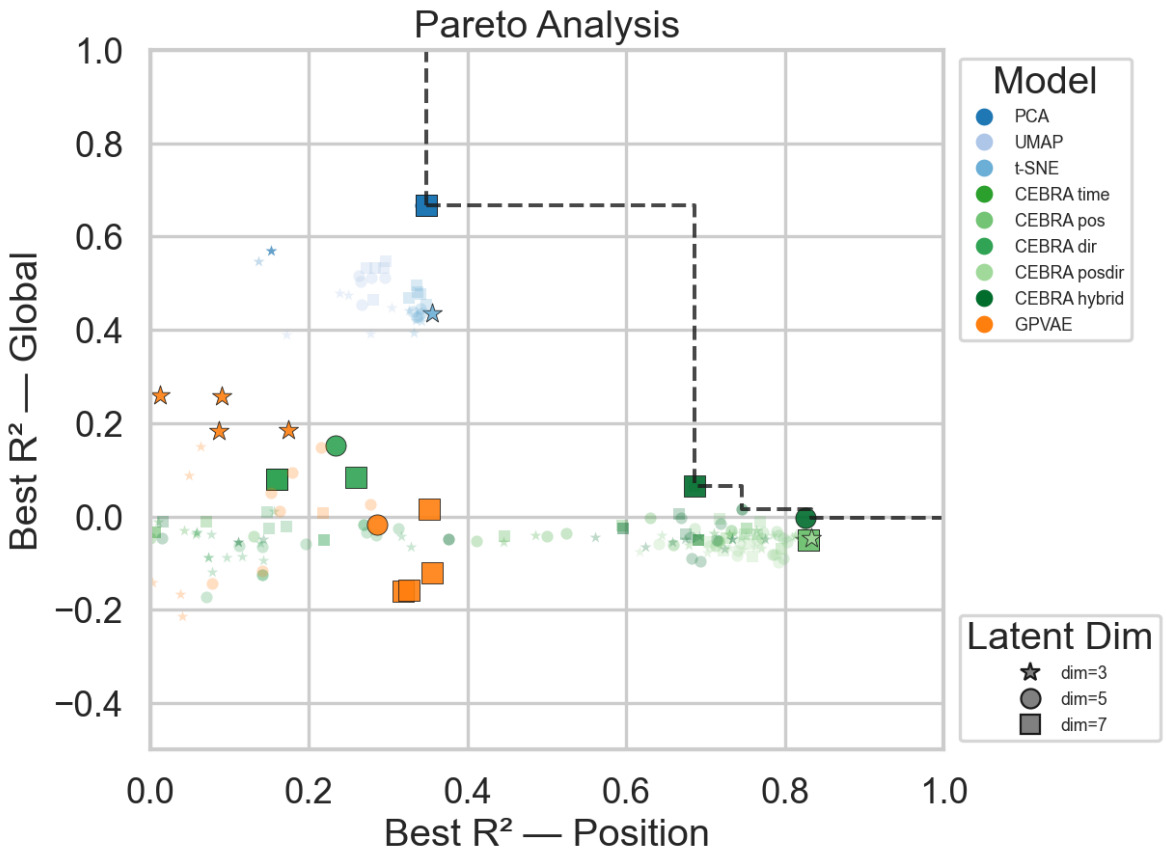

Figure 10: Pareto Front Analysis for non-linear R^2 scores for global signal reconstruction and position decoding in rat hippocampus dataset.

while maintaining moderate positional decoding performance. In contrast, CEBRA-Hybrid obtains the best positional decoding performance but at the cost of substantially lower global reconstruction. These results illustrate the fundamental trade-off between preserving the original structure of the neural activity and optimizing the latent representation for downstream behavioral decoding.

When both objectives are assigned equal importance, PCA emerges as the most balanced solution. Its superior reconstruction performance indicates that it preserves the original neural signal more faithfully than the other methods, while still retaining a reasonable amount of information relevant for position decoding. This explains why PCA ranks first in the benchmark despite being the simplest method evaluated.

The remaining CEBRA variants (CEBRA-Pos, CEBRA-Posdir, and CEBRA-Time) achieve high decoding scores but exhibit poor global reconstruction performance, causing them to move away from the Pareto front. Similarly, GP-VAE provides intermediate decoding performance but fails to preserve global structure effectively. UMAP and t-SNE occupy a middle ground, offering reconstruction scores above 0.5 while maintaining modest decoding performance.

As summarized in Table 4, the Pareto-optimal models at the per-model level are PCA and CEBRA-Hybrid, since no single model can improve one objective without sacrificing

the other.

Among all evaluated approaches, PCA achieves the highest global reconstruction score, while maintaining moderate positional decoding performance. In contrast, CEBRA-Hybrid obtains the highest positional decoding score among the table's per-model representatives, but at the cost of substantially lower global reconstruction. These results illustrate the fundamental trade-off between preserving the original structure of neural activity and optimizing the latent representation for downstream behavioral decoding.

When both objectives are assigned equal importance, PCA emerges as the most balanced solution. Its superior reconstruction performance indicates that it preserves the original neural signal more faithfully than any other evaluated method, while retaining a reasonable amount of information relevant for positional decoding. In addition, as illustrated in Figure 10, lower dimensional PCA variants, also achieve superior global reconstruction with respect the rest of LVMs. Accordingly, PCA achieves the smallest Euclidean distance to the ideal point $(1, 1)$ and ranks first under this balanced evaluation criterion, a notable result for the simplest method evaluated.

The remaining CEBRA variants (CEBRA-Pos, CEBRA-Posdir, and CEBRA-Time) achieve high decoding scores but exhibit very poor global reconstruction performance, pushing them away from the Pareto front. GP-VAE provides intermediate decoding performance but similarly fails to preserve global structure. CEBRA-Hybrid, which outperforms GP-VAE on both objectives. UMAP and t-SNE are both strictly dominated by PCA, which achieves higher scores on both metrics simultaneously; they therefore also fall off the Pareto front.

Rank	**Model**	R^2 **Pos.**	R^2 **Glob.**	**Distance to** $(1, 1)$
1	PCA	0.3831	0.7628	0.4450
2	CEBRA-hybrid	0.8280	0.1608	0.5375
3	CEBRA-Pos	0.7815	0.1047	0.5436
4	CEBRA-Posdir	0.7795	0.0930	0.6020
5	CEBRA-Time	0.7146	0.0370	0.6598
6	UMAP	0.3343	0.5503	0.6715
7	t-SNE	0.3451	0.5131	0.6746
8	GP-VAE	0.5595	0.1440	0.7327
9	CEBRA dir	0.2009	0.3839	0.7346

Table 4: Ranking of LVMs for the rat hippocampus benchmark based on positional decoding performance, global reconstruction performance and the Euclidean distance to the ideal point $(1, 1)$ in the two-objective space.

Overall, these results suggest that criteria based equally on decoding accuracy and signal preservation tend to favor classical linear methods such as PCA. Deep LVMs based on supervised representation-learning approaches, particularly CEBRA, are strongly optimized for behaviorally relevant information and therefore sacrifice aspects of the original neural geometry. Consequently, although these methods excel at decoding

tasks, they are penalized under a balanced multi-objective evaluation framework, preventing them from dominating the Pareto analysis. In Figure 10, this penalization is more evident, as CEBRA models concentrate close to Pareto front for positional decoding score, but stay below zero at global reconstruction score coordinate. However, the Pareto analysis reveals that CEBRA clearly outperforms the other models in terms of the efficiency and compactness with which positional information is encoded. Several CEBRA variants achieve high decoding performance using substantially fewer latent dimensions than competing methods, resulting in a cluster of low-dimensional solutions located near the Pareto front, suggesting that CEBRA is particularly effective at compressing behaviorally relevant information into compact latent representations.

5.2 Macaque Reaching Benchmark

In this second benchmark, the objective of the LVMs is to decode hand position from neural activity recorded during a monkey holding a manipulandum exposed to passive forces. Unlike the previous benchmark, the neural activity in this task is primarily proprioceptive, as the monkey do not exert any active force. Consequently, the recorded population activity exhibits greater variability and more complex dynamics, reflecting both movement execution and responses to external force perturbations.

The latent embeddings should reveal trajectories that closely resemble the hand movement patterns generated during the experiment (see Appendix A. Since the recorded neural populations encode the displacement induced by the external perturbations, successful latent representations tend to organize samples according to these movement trajectories. Therefore, the quality of the embeddings can be qualitatively assessed by the extent to which they preserve smooth and coherent trajectory structures. Only the position along a single axis is used for generating the color labels of the latent representations.

The barplots for non-linear decoders of positional decoding performance and global activity preservation are shown in Figure11 and Figure12, respectively. For positional decoding, PCA is the strongest baseline and matches or slightly trails the best supervised CEBRA variants. UMAP achieves only moderate scores, while t-SNE and GP-VAE largely fail at low latent dimensions, with GP-VAE partially recovering at highest dimension. For global activity reconstruction, results split sharply into models preserving neural activity with high fidelity across all tested dimensions and models scoring near or below zero.

5.2.1 Baseline Results

In Table 5, the optimal performance of the baseline LVM methods is summarized. PCA clearly outperforms the non-linear baselines, achieving both the highest positional decoding score and the highest global reconstruction score, both obtained at the maximum latent dimensionality. Notably, global reconstruction is already excellent at lowest dimension and remains stable across dimensions, while positional performance improves more consistently with increasing dimensionality. These results indicate that the dominant structure of neural activity can be effectively captured by a linear projection, and that hand position information is largely preserved in the principal components.

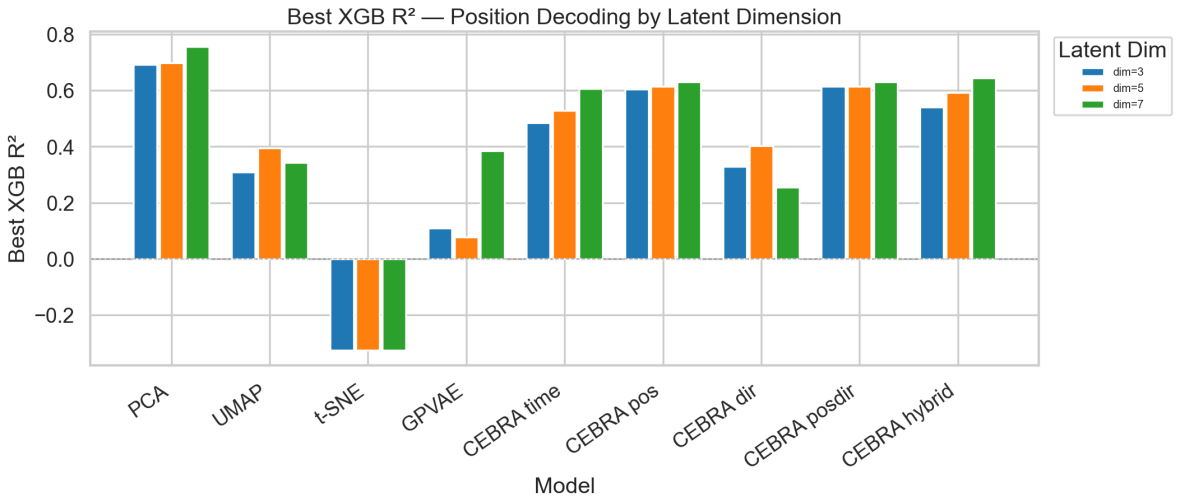

Figure 11: Barplot of the best R^2 scores for position decoding of each model variety using XGBoost estimators over latent dimensions for macaque reaching dataset.

In contrast, t-SNE exhibits the poorest overall performance. Although designed to preserve local neighborhood relationships, it strongly distorts the global geometry of the data, as the the non-linear decoder produces strongly negative value indicating that the learned embeddings actively harm reconstruction relative to a trivial predictor. Positional decoding is similarly poor, with scores ranging negative values depending on seed and dimensionality.

UMAP provides intermediate performance. It achieves moderate positional decoding, substantially outperforming t-SNE, but still fails to preserve the global neural activity structure, with both linear and XGBoost global scores near zero or negative. This suggests that while UMAP retains some behaviorally relevant information, it sacrifices much of the original signal geometry in the process.

Qualitatively, the latent embeddings support these quantitative findings. PCA produces smooth and organized trajectories that closely follow the movement structure of the holding task, consistent with its strong decoding and reconstruction performance. Conversely, UMAP and especially t-SNE generate more fragmented and disorganized latent trajectories. While some local trajectory patterns remain visible, the overall movement structure becomes less coherent, explaining their reduced ability to recover both position and global neural activity.

Due to the proprioceptive nature of neural activity, the signal manifests strong variability between activity states related to the position of the hand. Consequently, PCA, which extracts latent factors by maximizing explained variance, naturally recovers a representation that is linearly predictive of hand position, without requiring any behavioral labels. This results in PCA achieving higher positional decoding scores than the tested label-conditioned CEBRA variants. Importantly, this advantage does not extend to unsupervised non-linear methods: UMAP and t-SNE, also label-free, perform substantially worse, indicating that the success of PCA is tied specifically to the linear separability of position states in motor cortex, not to unsupervised learning in general.

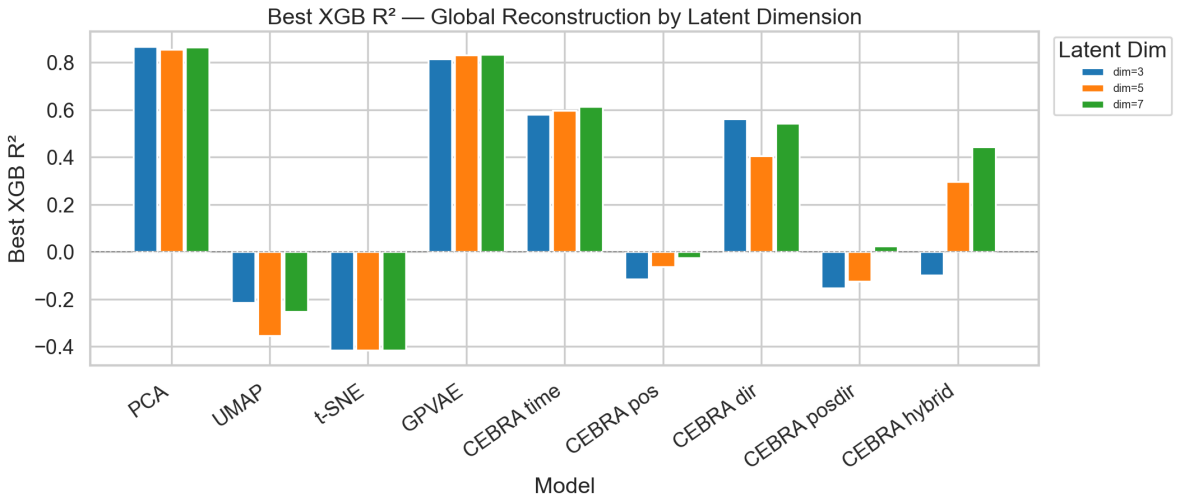

Figure 12: Barplot of the best R^2 scores for global activity reconstruction of each model variety using XGBoost estimators over latent dimensions for macaque reaching dataset.

Model	**Optimal Positional Decoding**		**Optimal Global Reconstruction**	
	R^2	**Latent Dim.**	R^2	**Latent Dim.**
PCA	0.7565 (xgb)	7	0.8933 (lin)	3
t-SNE	0.0430 (lin)	3	0.0149 (lin)	3
UMAP	0.4415 (lin)	5	0.1048 (lin)	3

Table 5: Comparison of baseline LVM methods optimal performance in macaque reaching dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin).

5.2.2 CEBRA Results

As shown in the barplots, CEBRA models generally improve positional decoding with increasing latent dimensionality, while already achieving substantial performance at minimal dimensions. An exception is CEBRA-Dir, whose positional decoding is unstable across dimensions and remains substantially lower than the other variants. Unlike previous case, CEBRA models substantially improve the positional decoding with higher dimensions, but still manifest decent performance at lowest.

The learned embeddings capture meaningful positional structure, but are less visually and geometrically interpretable than in previous benchmarks due to the increased variability and ambiguity of the behavioral labels. This particularly affects methods that rely strongly on positional supervision. In contrast, CEBRA variants that incorporate weaker or alternative inductive biases, such as CEBRA-Time and CEBRA-Dir, are less constrained by noisy positional labels and therefore generalize more robustly.

Notably, as reported in Table 6, CEBRA-Time achieves competitive and balanced performance across both positional decoding and global reconstruction. This indicates that temporal contiguity supervision, which does not directly enforce positional clustering, is sufficient to recover behaviorally relevant structure while also preserving a significant

Model	Optimal Positional Decoding			Optimal Global Reconstruction		
	R^2	Latent Dim.	Time Offset	R^2	Latent Dim.	Time Offset
CEBRA time	0.6073 (xgb)	7	40	0.7170 (lin)	7	10
CEBRA pos	0.6310 (xgb)	7	40	0.1631 (lin)	7	50
CEBRA dir	0.5159 (lin)	5	50	0.6247 (lin)	3	20
CEBRA posdir	0.6304 (xgb)	7	10	0.1968 (lin)	7	30
CEBRA hybrid	0.6437 (xgb)	7	20	0.5885 (lin)	7	50

Table 6: Comparison of baseline LVM methods optimal performance in macaque reaching dataset. Highest R^2 score of the two decoders is selected: non-linear (xgb) and linear (lin).

portion of the global signal geometry. In contrast, CEBRA-Dir performs poorly on positional decoding, showing that movement direction alone is not sufficient to recover precise hand position in this task.

The violin plots (available in Appendix B) show the performance of the different CEBRA models over dimensions. CEBRA-Time now achieves meaningful positive global reconstruction (0.35–0.50), improving with dimension, as temporal contrastive learning captures better genuine motor dynamics. The position/global trade-off persists for position-supervised variants, as position decoding is good but global remains negative. CEBRA-Dir, unlike in previous case, yields positive and growing global reconstruction with dimension, indicating that movement direction aligns naturally with the population geometry of motor cortex. CEBRA-Hybrid shows the strongest dimensionality dependence, as both position decoding and global reconstruction improve monotonically from dim=3 to dim=7, suggesting the model requires more capacity to simultaneously satisfy both objectives.

Conversely, it is reflected clearly in the global reconstruction scores, as CEBRA-Time and CEBRA-Dir preserve moderate-to-good global structure, while CEBRA-Pos and CEBRA-Posdir achieve strong position decoding at the cost of strongly negative global score. This trade-off is a direct consequence of the contrastive objective, by pulling together trials sharing the same positional label, the embedding collapses variance unrelated to position, thereby degrading global reconstruction. CEBRA-hybrid partially mitigates this, achieving the best position score with a less severe global penalty.

Consequently, CEBRA's position-supervised variants are able to decode trajectories robustly despite the complexity of the motor task, but this comes at the cost of global activity preservation, a trade-off driven by the contrastive objective rather than signal variability.

5.2.3 GP-VAE Results

For GP-VAE, the results indicate a clear asymmetry between reconstruction and decoding performance. Within the evaluated latent dimensionality range, the best positional decoding score is $R^2 = 0.4335$ (linear decoder, $d = 7$) and the best global reconstruc-

tion score is $R^2 = 0.8854$ (linear decoder, $d = 7$). Both optima are achieved with low regularization: `beta` $= 0.1$ and `lambda_contrastive` $= 0.1$, the smallest values tested.

This suggests that stronger variational and contrastive regularization is detrimental in this setting. The GP-VAE contrastive term penalizes differences between adjacent temporal windows, effectively adding smoothness constraints on top of the Gaussian process prior. When `lambda_contrastive` is large, this redundant constraint over-regularizes the latent space and prevents the model from capturing detailed structure in the neural dynamics, degrading both decoding and reconstruction performance.

The violin plot of GP-VAE is illustrated in Figure B.11, which shows that GP-VAE is highly sensitive to all tested hyperparameters. Position decoding improves monotonically with latent dimension, but global reconstruction remains poor and volatile throughout, as the GP prior is insufficient for the complex hippocampal dynamics. These sensitivities make GP-VAE considerably harder to tune than CEBRA for this dataset.

Regarding global reconstruction, barplots and violin plots (see Figure B.11 at Appendix B.) demonstrate that GP-VAE is highly efficient even at lowest dimension, with performance remaining consistently high at lower dimensions. This indicates that the smooth, low-dimensional structure of motor cortex population activity during reaching is well captured by the GP-VAE's Gaussian latent prior with minimal dimensionality. However, for positional decoding the model struggles to compress information, as at lower dimensions the distribution is centered near zero with high variance and frequent negative values, indicating that most hyperparameter configurations fail entirely. At highest dimensionality, the distribution shifts upward and the best-case performance approaches. but the variance remains large and the median stays low. This means GP-VAE requires higher latent dimensionality to have any chance of partially encoding hand position, and even then performance is unreliable and substantially below PCA. The model therefore cannot efficiently compress positional information regardless of the dimensionality used, unlike global activity, which it compresses well even at the lowest tested dimension.

Qualitatively, GP-VAE embeddings preserve a smooth, approximately Gaussian latent structure reflecting the temporal continuity of motor dynamics, while only partially separating samples by positional label. This separation is far less defined than in models explicitly optimized for behavioral decoding, consistent with the trade-off between probabilistic temporal structure and movement-specific discrimination.

5.2.4 Pareto Analysis

After comparing the performance for both metrics, PCA clearly dominates the benchmark. It achieves the highest positional decoding score and the highest global reconstruction score, resulting in the smallest distance to the ideal point $(1, 1)$. Consequently, PCA is the only model located on the Pareto front, as no other method surpasses it in either objective. In practice, it defines the optimal trade-off between positional decoding and signal preservation in this benchmark.

Despite this outstanding quantitative performance, PCA should not automatically be

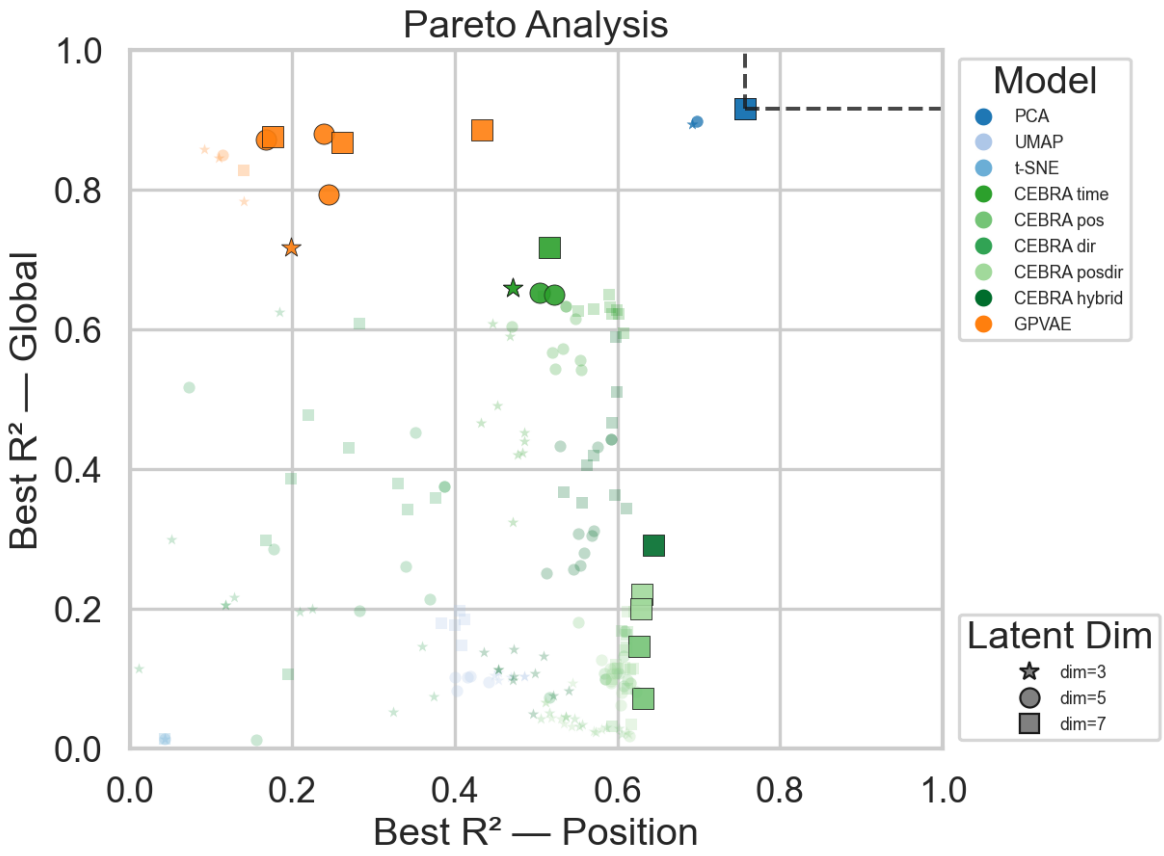

Figure 13: Pareto Front Analysis for non-linear R^2 scores for global signal reconstruction and position decoding in macaque reaching dataset.

considered the most interpretable representation. Its optimal performance is obtained with a relatively large latent dimensionality, allowing the model to capture a substantial fraction of the neural variability. While this improves both reconstruction and decoding accuracy, the resulting representation distributes information across many components, making it harder to identify compact task-specific latent factors.

Among the non-linear approaches, CEBRA-Time provides the strongest compromise between reconstruction and decoding performance. It achieves an excellent positional decoding score while preserving global neural activity with a moderate reconstruction score, yielding the second-highest overall ranking. GP-VAE follows closely at rank 3. Although its positional decoding performance is lower, it achieves a remarkably high global reconstruction score, substantially outperforming all CEBRA variants in this objective. This suggests that the temporal Gaussian Process prior is particularly effective at preserving the global structure of neural activity.

CEBRA-Hybrid occupies the fourth position, achieving competitive positional decoding but a more modest reconstruction score. This indicates that combining multiple supervisory signals improves behavioral decoding but does not necessarily preserve the underlying neural dynamics. Similarly, CEBRA-Pos and CEBRA-Posdir obtain relatively strong positional decoding scores, yet exhibit poor reconstruction performance, suggesting that strong positional supervision biases the embedding toward behavioral

information at the expense of signal fidelity.

Interestingly, CEBRA-Dir performs differently from the other supervised variants. Although its positional decoding score is comparatively low , it preserves a substantial amount of global neural activity. This result indicates that directional supervision captures aspects of the neural dynamics that are more closely related to the global structure of the signal than precise positional labels.

UMAP and especially t-SNE perform poorly in both objectives. UMAP achieves only moderate positional decoding and limited reconstruction performance, while t-SNE remains near zero for both metrics. These results suggest that neighborhood-preserving manifold learning techniques are less suitable for capturing either the behavioral information or the global neural structure present in this dataset.

Rank	**Model**	R^2 **Pos.**	R^2 **Glob.**	**Best Latent Dim**	**Distance to** $(1,1)$
1	PCA	0.7565	0.9152	7	0.258
2	CEBRA time	0.5899	0.6493	7	0.540
3	GP-VAE	0.4335	0.8854	7	0.578
4	CEBRA hybrid	0.5966	0.3631	7	0.754
5	CEBRA dir	0.2823	0.6089	7	0.817
6	CEBRA posdir	0.6304	0.2213	7	0.862
7	CEBRA pos	0.6268	0.1460	7	0.932
8	UMAP	0.4113	0.1851	7	1.005
9	t-SNE	0.0430	0.0143	7	1.374

Table 7: Ranking of latent variable models for the macaque reaching benchmark using only latent dimensions, based on positional decoding performance, global reconstruction performance, and the Euclidean distance to the ideal point $(1, 1)$ in the two-objective space.

A key factor underlying the performance differences across models is their sensitivity to the number of latent dimensions. PCA exhibits strong dimensionality dependence, as its optimal performance is reached at the maximum dimensionality considered ($d = 7$), where enough principal components are retained to simultaneously explain the variance required for position decoding and global reconstruction. At lower dimensionalities, positional decoding performance degrades substantially, illustrating the classical information loss associated with aggressive dimensionality reduction. Nevertheless, PCA remains capable of preserving a large proportion of the global activity signal even at relatively low dimensionalities, indicating that the dominant variance of the neural population can be represented efficiently with only a few components.

Similarly, GP-VAE achieves its best performance at the highest latent dimensionality. Its probabilistic temporal prior provides structured regularization that enables the model to preserve the overall organization of the neural activity while compressing the signal into a compact latent space. This inductive bias explains why GP-VAE achieves the second-highest reconstruction score despite its moderate decoding performance.

However, additional latent dimensions appear necessary to capture the behavioral information required for accurate position decoding.

The CEBRA variants exhibit a more nuanced dimensionality profile. CEBRA-Time benefits from higher dimensionalities, where temporal supervision helps organize the latent space while preserving meaningful aspects of the original signal. In contrast, CEBRA-Pos and CEBRA-Posdir primarily exploit additional dimensions to improve behavioral decoding, with little corresponding gain in reconstruction quality. This behavior reinforces the notion that strong behavioral supervision encourages the model to prioritize task-relevant information over complete representation of neural activity.

UMAP exhibits only marginal improvement with increasing dimensionality and quickly reaches a performance plateau, reflecting the fact that its topology-preserving objective is not directly aligned with either decoding performance or signal reconstruction. Likewise, t-SNE remains largely insensitive to the number of latent dimensions, with its performance being dominated by stochastic optimization effects rather than representational capacity.

Overall, the Pareto analysis reveals that preserving global neural dynamics is strongly associated with overall performance in this benchmark. While PCA remains the best-performing method, GP-VAE emerges as the strongest non-linear alternative for preserving neural activity structure, whereas CEBRA-Time provides the best balance between behavioral decoding and reconstruction among the supervised approaches. These results highlight the fundamental trade-off between encoding task-relevant information and maintaining fidelity to the original neural signal.

6 Execution schedule

This section describes the planning and scheduling followed to develop this project, including the different tasks required for its completion and their temporal organization. The aim of this plannification is to ensure that the project is carried out efficiently, assigning sufficient time to each stage while maintaining realistic deadlines and accounting for potential difficulties that may arise during development.

6.1 Work Breakdown Structure (WBS)

To facilitate the development of the project, the workload was divided into six technical phases. These phases constitute the fundamental pillars of the project, providing a clear organizational structure and establishing a systematic execution workflow throughout the study. To identify, organize and manage the different activities within the project's scope, a WBS methodology was adopted.

Activity ID	**Activity / Task Description**	**Precedents**	**Duration (Hours)**
A	1.1 Planification of Objectives, Resources, and Timelines	-	10
B	1.2 Market and Feasibility Analysis	A	10
C	1.3 Regulatory and Legal Compliance Analysis	A	5
D	2.1 Computational Neuroscience Foundations	A	50
E	2.2 Baseline LVM framework	D, B	10
F	2.3 Non-generative deep LVM framework	D, B	20
G	2.4 Generative deep LVM framework	D, B	20
H	3.1 Rat Hippocampus Dataset Loading and Description	C	25
I	3.2 Monkey Reaching Dataset Loading and Description	C, H	25
J	4.1 Design and Implementation of a Standard Pipeline	E, F, G, H	35
K	4.2 Baseline LVMs Implementation	J	30
L	4.3 CEBRA Implementation	J	30
M	4.4 GP-VAE Implementation	J	35
N	4.5 Results Extraction and Visualization	K, L, M	20
O	5.1 Performance Analysis in Rat Hippocampus Dataset	N	15
P	5.2 Performance Analysis in Monkey Reaching	N	15
Q	5.3 Project Conclusions and Future Outlook	O, P	15
R	6.1 Draft Compilation and Writing	Q	15
S	6.2 Document Review, Editing and Formatting	R	10
T	6.3 Thesis Defense Preparation	S	5

Table 8: Work Breakdown Structure (WBS) activities, precedence relationships, and estimated durations for the project.

As shown in Table 8, the WBS decomposes the project into smaller and more manageable work packages, enabling a structured allocation of tasks and resources. In this

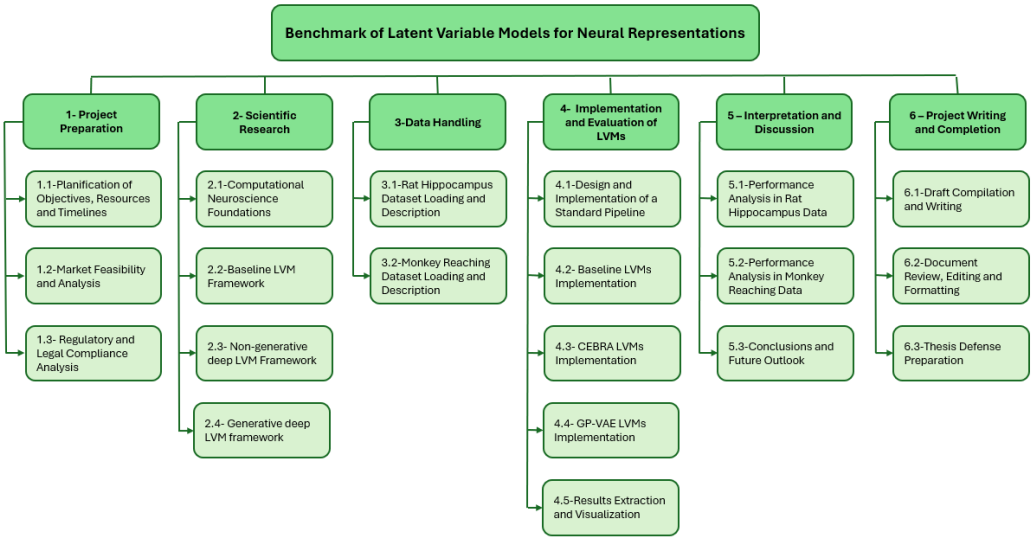

Figure 14: WBS diagram illustrating the technical phases of the project

project, the WBS is organized into six main work packages: project preparation, scientific research, data handling, implementation and evaluation of LVMs, interpretation and discussion of results, and project writing and completion. Figure 14 presents the WBS developed for this study, providing an overview of the main work packages and their associated tasks.

6.2 Execution Chronogram (GANTT)

To complement the WBS and provide additional detail on the temporal progression of the project, a Gantt chart is implemented, as shown in Figure 15. This type of diagram enables the visualization of project progression.

6.3 Critical Path Method Diagram (PERT)

Complementary to the Gantt diagram, the PERT diagram also provides a visual representation of the chronology and dependencies between activities, as illustrated in Figure 16. In this diagram, activities are represented as arrows. The critical path, which highlights the sequence of tasks with no scheduling flexibility and that must be completed on time to avoid delays in the overall project, is shown in red. In contrast, non-critical activities, which have scheduling flexibility and can tolerate delays without affecting the overall project timeline, are displayed in black.

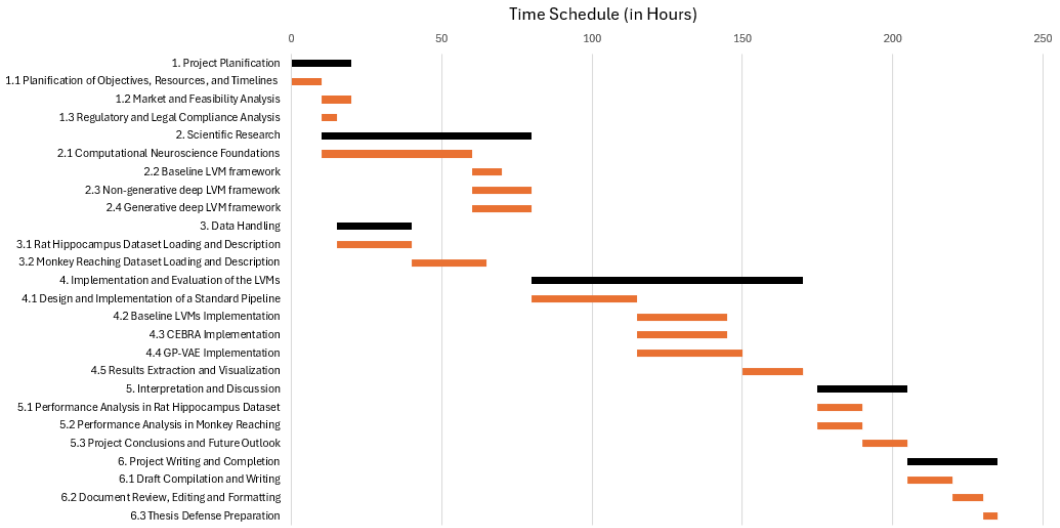

Figure 15: GANTT chart illustrating the timeline progression of the project

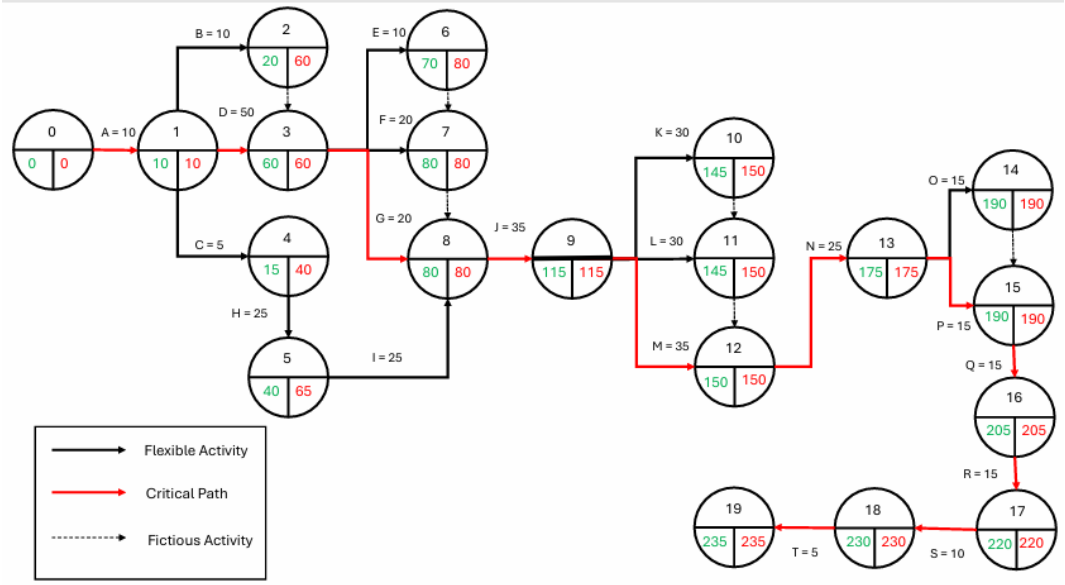

Figure 16: WBS diagram illustrating the technical phases of the project

7 Technical viability

In this section, the technical viability is explored through SWOT analysis. In particular, strengths, weaknesses, opportunities and threats regarding technical aspects of the project are identified to evaluate the internal and external factors that could influence the project's execution from a technological standpoint.

	Helpful	Harmful
Internal	**Strengths** • Access to server infrastructure (CUDA, tmux sessions) • Predefined and validated LVMs • High-quality, preprocessed neural datasets with behavioral labels • Modular and reproducible benchmarking pipeline • Extensive institutional and bibliographic support • Ready-to-use evaluation and visualization tools	**Weaknesses** • High interdisciplinary complexity • Technically demanding frameworks • Difficulty interpreting latent representations beyond metrics • Limited dataset diversity only exploring specific behaviors • Computational cost for multimodel and hyperparameter training • Benchmark sensitivity to decoder choice and setup
External	**Opportunities** • Expansion to new neural datasets and behavioral tasks • Inclusion of new LVM architectures • Development of improved evaluation metrics for embeddings • Potential clinical applications in BCIs	**Threats** • Ethical and regulatory constraints on neural data collection • Strong dependence on external software and updates • Barriers to clinical translation

Table 9: SWOT analysis

The strengths section defines internal resources and competences that this project may leverage to its advantage. Regarding these factors, computationally demanding models can be trained and evaluated in significantly reduced execution times thanks to the available computational infrastructure. In addition, the use of `tmux` sessions enables long-running simulations to be executed and monitored reliably without interrupting the experimental workflow. The benchmark models are already available and validated, and the selected datasets are preprocessed and integrated within the CEBRA package, providing high-quality neural recordings and behavioral labels. Furthermore, researchers and collaborators with expertise in computational neuroscience provide valuable guidance for understanding abstract concepts from the bibliographic resources, interpreting results and making methodological decisions throughout the development

process. The modular structure of the benchmark also benefits from intuitive evaluation and visualization tools, facilitating a clearer understanding of the performance of the tested LVMs.

In contrast, the weaknesses section targets internal limitations for conducting the benchmark. First, the project requires expertise across multiple domains, including computational neuroscience, machine learning, and statistics, which can slow down the learning process and the assimilation of complex concepts. Additionally, implementing and adapting several models requires familiarity with specialized frameworks such as PyTorch and Pyro, as well as experience with CUDA environments, remote servers, Bash scripting, and tmux session management, making the project more prone to implementation errors. The benchmark focuses on two datasets involving exclusively positional information. While these datasets provide interpretable behavioral labels, conclusions regarding the performance of the evaluated LVMs may not fully generalize to other neural recording modalities, brain regions, or behavioral tasks. Furthermore, although hyperparameter exploration is possible, training multiple configurations of GP-VAE and CEBRA across different latent dimensions and datasets remains time-consuming even with available computational resources. The choice of fixed decoder configurations may influence the measured performance of the models, as differences in decoding ability can be partially affected by decoder hyperparameters rather than solely reflecting the quality of the learned latent representations.

For exploring the growth of the presenting project, opportunities part describes those external factors that may enhance the technical viability of the project. The field of computational neuroscience is experiencing rapid growth, with increasing interest in understanding how neural activity can be represented in low-dimensional latent spaces. This creates strong opportunities for the benchmark to contribute to an active and evolving research area in representation learning and neural dynamics modeling. Advances in neural recording technologies are generating increasingly large and diverse datasets. In this context, the benchmark can be extended to evaluate latent representations learned from multimodal recordings, larger neural populations, or cross-subject datasets, thereby improving its generality and scientific relevance. Furthermore, emerging LVMs architectures can be easily integrated into the benchmarking framework due to its modular design. Similarly, the framework can be extended with new evaluation metrics that capture additional properties of embeddings, such as geometry or topology, beyond standard decoding performance. These methods also open the possibility of future applications in neural decoding for clinical contexts, including BCIs, where latent representations could contribute to the interpretation and decoding of neural activity in human patients.

Threats analysis focus on external factors hindering project's technical viability and expansion. The acquisition of high-quality neural recordings often relies on invasive experimental *in vivo* procedures, as LVMs require behavior context to evaluate their decoding performance. Consequently, ethical considerations and increasingly strict regulations governing animal experimentation and invasive neural recording techniques may limit the availability of new datasets, restricting future extensions and validation of the benchmark. In addition, some LVMs and associated software implementations are still under active development and may undergo periodic updates or change, which may

cause compatibility issues, unexpected behavior or reproducibility challenges within the benchmarking pipeline. Although many LVM techniques show promising potential for applications in BCI systems, their translation to clinical environments is subject to strict regulatory requirements.

8 Economic viability

This document establishes the financial and resource framework for the project. The budget is divided into three fundamental pillars: human resources, computational infrastructure and energy, and material and data resources.

First, regarding human resources, the project team consists of a biomedical engineer researcher and a supervising tutor. The labor costs are calculated based on competitive salary references for their respective professional roles and hours of dedication [74]. However, as the biomedical engineer is an undergraduate with limited experience in the field, the salary was slightly reduced to a rate of 20.00 €/hour for the 400-hour duration of the project. Conversely, the academic and technical tutoring provided by the supervisor, comprising 50 hours, was estimated to a rate of 30.00 €/hour. Therefore, the subtotal human resources arises from the addition of these two work rates:

The modeling workflows were executed on a pre-purchased, fully configured workstation node managed by the Department of Biomedicine. Because this hardware asset was already acquired and mounted prior to the project's initiation, it represents a 0.00 € capital expenditure. To perform model simulations, the architecture operated continuously 24/7 across the entire 4-month duration of the project. Operating under a steady low active workload, the system maintained a highly controlled, non-excessive power draw averaging 0.76 kW [75]. Considering the total power consumed during the operational window of four months, $2,225.28$ kW were consumed. The regional power utility index is fixed at the standardized European Union operational average rate of 0.29 €/kW for medium-sized institutional consumers [76]. Therefore, the total operational energy cost estimates to 645.33 €.

Algorithmic base frameworks and validation datasets are fully open-source and publicly licensed, introducing zero data acquisition friction or premium database fees. All advanced bibliographic databases, journals, institutional reviews, and scientific assets were systematically accessed through standard library subscriptions provided by the University of Barcelona.

Budget Category	**Total Cost**
1. Human Resources	7,500.00 €
2. Technical Infrastructure	0.00 €
3. Energy Consumption	645.33 €
4. Data and Literature	0.00 €
Total Project Budget	**8,145.33 €**

Table 10: Summary project's budget

9 Regulatory and legal aspects

This section outlines the principal legal and regulatory considerations relevant to the project. Although these aspects do not directly constrain the technical objectives of the benchmark, they are closely related to the origin of the data used for model development and to the potential future deployment of LVMs in BCI systems involving human subjects. Consequently, the regulatory framework encompasses both animal research governance and medical device regulation.

The datasets employed in this benchmark originate from experiments conducted on animal subjects. As a result, the acquisition of the original neural recordings is subject to strict ethical and legal requirements designed to ensure the ethical treatment of animals used in scientific research. Modern animal research governance is largely based on the principles of replacement, reduction, and refinement (3Rs) [77]. These principles establish that researchers should employ alternative methods whenever possible, such as computational simulations, synthetic data, or organoid models (replacement); minimize the number of animals required to obtain scientifically valid results (reduction); and implement procedures that reduce pain, distress, and long-term harm to the animals involved (refinement).

All the datasets analyzed in this project were collected in the United States, where animal experimentation is regulated primarily through the Animal Welfare Act (AWA) and the Public Health Service (PHS) Policy on Humane Care and Use of Laboratory Animals, administered by the United States Department of Agriculture (USDA) and the National Institutes of Health (NIH), respectively [78], [79]. Research institutions conducting animal studies must maintain an approved Institutional Animal Care and Use Committee (IACUC), which is responsible for reviewing and approving experimental protocols before any data collection takes place [80]. Such protocols include surgical procedures, recording methodologies, monitoring procedures, and post-operative care plans.

Although this project only involves the secondary use of previously collected data, it remains important to ensure that the datasets originate from studies conducted under approved ethical protocols. Scientific journals, public repositories, and funding agencies commonly require explicit documentation of ethical approval as a prerequisite for publication and data sharing. Consequently, compliance with established animal welfare regulations remains relevant even when neural recordings are reused exclusively for computational analysis.

Beyond the acquisition of the data itself, an additional regulatory dimension arises from the potential application of LVMs in future BCI systems. When ML algorithms are integrated into clinical BCIs, they may fall under the category of Software as a Medical Device (SaMD) [81]. In these applications, LVMs can be used to transform neural activity into control signals for external devices, such as prosthetic limbs, communication systems, or assistive technologies.

Within the European Union, software components used in BCI systems are generally regulated under the Medical Device Regulation (EU MDR 2017/745). Depending on the

intended clinical application and associated risk profile, such systems may be classified as Class IIb or Class III medical devices [82]. Obtaining market approval requires CE marking, supported by extensive evidence demonstrating safety, clinical performance, reliability, and reproducibility in human populations.

A major challenge in the regulatory assessment of ML-based BCIs concerns the transparency and reliability of the underlying algorithms. Most deep LVMs relying on large neural network architectures, are often characterized as black-box systems because their internal decision-making processes are difficult to interpret [83]. Consequently, regulatory authorities may require additional evidence regarding model behavior, robustness and failure modes before such systems can be approved for clinical use.

Several challenges must be addressed before models developed using animal neural recordings can be safely translated to human applications. It is necessary to demonstrate that the learned representations generalize reliably to human neural activity, including neurological conditions, such as spinal cord injury or amyotrophic lateral sclerosis [84]. The long-term stability of latent representations must be validated to prevent performance degradation or representational drift that could compromise device safety.

Although the present project does not involve human experimentation or clinical deployment, understanding these regulatory and ethical considerations is essential for contextualizing the broader impact of LVMs and their potential future integration into clinical neurotechnology systems.

10 Conclusions and future outlook

PCA emerged as the highest-performing method according to the Pareto analysis, across both benchmark datasets. In the rat navigation dataset, PCA achieved the best compromise between positional decoding and reconstruction quality, while in the macaque reaching dataset it clearly dominated all competing approaches. These results indicate that a large fraction of the information relevant for both behavior and neural activity reconstruction is contained within linear projections of the population activity. However, this does not necessarily imply that PCA is the most suitable model for these tasks. Its superior performance was generally achieved using relatively high-dimensional latent spaces, which allow the model to capture a large portion of the variability present in the original signal. In contrast, several CEBRA variants achieved substantially higher positional decoding performance while relying on far fewer latent dimensions. This ability to compress behaviorally relevant information into compact latent representations constitutes an important advantage over baseline methods, as it reduces computational cost, improves efficiency, and facilitates interpretation of the learned embeddings. Therefore, although PCA achieved the best overall scores, its latent representations are not necessarily the most compact or informative from a representation-learning perspective, and further inspection should be conducted to asses the ability to compact behavioral and neural information in deep LVMs with respect the baselines.

In contrast, supervised deep LVMs produced embeddings that were generally more structured and behaviorally interpretable. Models such as CEBRA explicitly use external variables to shape the latent manifold, resulting in representations where task-relevant information is easier to visualize and interpret. This effect was particularly evident in the rat navigation benchmark, where the behavioral labels followed smooth spatial trajectories and the learned embeddings closely reflected the underlying environment structure. However, these methods rely heavily on the quality and consistency of the supervisory signal. In the monkey reaching dataset, position constituted a considerably more variable behavioral variable, and the CEBRA variants that relied primarily on positional supervision exhibited a clear reduction in reconstruction performance. The increased variability of the labels prevented the model from learning latent spaces that generalized consistently across trials, particularly for global activity reconstruction. These results suggest that the supervised framework of CEBRA is highly dependent on the structure and informativeness of the auxiliary signal, which may limit its generalization capabilities when supervision is noisy or poorly structured.

Nevertheless, the benchmark highlights an important limitation of supervised approaches. By optimizing the latent space with respect to external labels, these models inevitably prioritize behaviorally relevant information at the expense of preserving other aspects of the neural activity. As a result, they often achieve superior decoding performance while exhibiting lower reconstruction accuracy. This trade-off is reflected in the Pareto analyses, where several CEBRA variants achieved excellent positional decoding but failed to preserve the original signal as effectively as PCA. Although this behavior may be considered a limitation under the evaluation criteria adopted in this work, it is also a consequence of the model's objective, as it maximizes the representation of behaviorally relevant information rather than preserving all neural variability equally.

However, the hyperparameter exploration performed in this work was necessarily limited by computational resources and time constraints. Several potentially influential parameters, including temporal window length, kernel configurations, and alternative regularization schemes, were not exhaustively evaluated and may lead to improved performance in future studies. This limitation is particularly relevant for GP-VAE. The temporal window size, which is one of the most influential hyperparameter, was not systematically explored despite being directly related to the model's ability to capture temporal dependencies in neural activity. As observed in the experimental results, the performance of GP-VAE was below the expectations reported in previous studies. It is therefore plausible that a more extensive hyperparameter search could substantially improve the quality of the learned representations. In contrast, this issue is less critical for CEBRA, since many of its hyperparameters were selected following the recommendations provided in the official documentation, which are designed to offer near-optimal configurations across a wide range of applications. For future extensions of this work, a more exhaustive exploration of GP-VAE would be particularly valuable, as the results suggest that combining temporal Gaussian process priors with supervised objectives can be a viable direction for neural representation learning.

For future extensions of this work, the benchmark could be improved by incorporating additional evaluation metrics. The current global reconstruction metric, based on the mean population activity, may overlook neuron-specific variability and potentially favour methods such as PCA, which are designed to maximize variance preservation. Alternative measures that better capture the full structure of the neural signal, including neuron-wise variability, information-theoretic quantities such as entropy and topological or geometric properties of the latent space, could provide a more comprehensive assessment of representation quality.

Finally, studies of this kind represent an important step toward the development of reliable BCIs and neural decoding systems. Understanding the strengths and limitations of different LVMs facilitates the selection of representations best suited to specific scientific and clinical objectives. Although no single method emerged as universally optimal, further research is needed to determine which approaches are most effective under different conditions. The results show that different models occupy distinct positions along the trade-off between interpretability, information preservation, and behavioral decoding performance. Characterizing these trade-offs is essential for advancing both fundamental neuroscience research and the development of future neurotechnology applications.

11 Bibliography

References

[1] W. E. Allen et al., "Thirst regulates motivated behavior through modulation of brainwide neural population dynamics," *Science*, vol. 364, no. 6437, eaav3932, 2019.

[2] S. Vyas, M. D. Golub, D. Sussillo, and K. V. Shenoy, "Computation through neural population dynamics," *Annual review of neuroscience*, vol. 43, no. 1, pp. 249–275, 2020.

[3] M. W. Mathis, A. P. Rotondo, E. F. Chang, A. S. Tolias, and A. Mathis, "Decoding the brain: From neural representations to mechanistic models," *Cell*, vol. 187, no. 21, pp. 5814–5832, 2024.

[4] J. P. Cunningham and B. M. Yu, "Dimensionality reduction for large-scale neural recordings," *Nature neuroscience*, vol. 17, no. 11, pp. 1500–1509, 2014.

[5] A. E. Urai, B. Doiron, A. M. Leifer, and A. K. Churchland, "Large-scale neural recordings call for new insights to link brain and behavior," *Nature neuroscience*, vol. 25, no. 1, pp. 11–19, 2022.

[6] C. Stringer and M. Pachitariu, "Analysis methods for large-scale neuronal recordings," *Science*, vol. 386, no. 6722, eadp7429, 2024.

[7] M. Sahani, *Latent variable models for neural data analysis*. California Institute of Technology, 1999.

[8] F. Pei et al., "Neural latents benchmark'21: Evaluating latent variable models of neural population activity," *arXiv preprint arXiv:2109.04463*, 2021.

[9] S. Schneider, J. H. Lee, and M. W. Mathis, "Learnable latent embeddings for joint behavioural and neural analysis," *Nature*, 2023. DOI: `https://doi.org/10.1038/s41586-023-06031-6`.

[10] M. Dabagia, K. P. Kording, and E. L. Dyer, "Aligning latent representations of neural activity," *Nature Biomedical Engineering*, vol. 7, no. 4, pp. 337–343, 2023.

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

[12] R. Gondur, U. B. Sikandar, E. Schaffer, M. C. Aoi, and S. L. Keeley, "Multimodal gaussian process variational autoencoders for neural and behavioral data," *arXiv preprint arXiv:2310.03111*, 2023.

[13] B. Ros Blanco, "Non-linear representation learning of neuronal population dynamics," M.S. thesis, Universitat de Barcelona, Facultat de Física, 2025.

[14] P. Gupta, N. Balasubramaniam, H.-Y. Chang, F.-G. Tseng, and T. S. Santra, "A single-neuron: Current trends and future prospects," *Cells*, vol. 9, no. 6, p. 1528, 2020.

[15] M. Azabou, K. X. Pan, V. Arora, I. J. Knight, E. L. Dyer, and B. A. Richards, "Multi-session, multi-task neural decoding from distinct cell-types and brain regions," in *The Thirteenth International Conference on Learning Representations*, 2025.

[16] M. G. Perich, D. Narain, and J. A. Gallego, "A neural manifold view of the brain," *Nature Neuroscience*, vol. 28, no. 8, pp. 1582–1597, 2025.

[17] M. R. Cohen and J. H. Maunsell, "A neuronal population measure of attention predicts behavioral performance on individual trials," *Journal of Neuroscience*, vol. 30, no. 45, pp. 15 241–15 253, 2010.

[18] B. Ros, M. Olives-Verger, C. Fuses, J. M. Canals, J. Soriano, and J. Abante, "Integration of calcium imaging traces via deep generative modeling," in *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2026, pp. 6986–6990.

[19] N. A. Steinmetz, C. Koch, K. D. Harris, and M. Carandini, "Challenges and opportunities for large-scale electrophysiology with neuropixels probes," *Current opinion in neurobiology*, vol. 50, pp. 92–100, 2018.

[20] W. Zong et al., "Large-scale two-photon calcium imaging in freely moving mice," *Cell*, vol. 185, no. 7, pp. 1240–1256, 2022.

[21] B. Drukarch and M. M. Wilhelmus, "Thinking about the action potential: The nerve signal as a window to the physical principles guiding neuronal excitability," *Frontiers in cellular neuroscience*, vol. 17, p. 1 232 020, 2023.

[22] M. E. J. Obien, K. Deligkaris, T. Bullmann, D. J. Bakkum, and U. Frey, "Revealing neuronal function through microelectrode array recordings," *Frontiers in neuroscience*, vol. 8, p. 423, 2015.

[23] M. Alaylioğlu, E. Dursun, S. Yilmazer, and D. G. Ak, "A bridge between in vitro and in vivo studies in neuroscience: Organotypic brain slice cultures," *Archives of neuropsychiatry*, vol. 57, no. 4, p. 333, 2020.

[24] Y. G. Maguire et al., "Physical principles for scalable neural recording," *Frontiers in computational neuroscience*, vol. 7, p. 61 935, 2013.

[25] J. Rettinger, S. Schwarz, and W. Schwarz, *Electrophysiology*. Springer, 2022.

[26] S. L. Hooper and J. Schmidt, "Electrophysiological recording techniques," *Neurobiology of Motor Control: Fundamental Concepts and New Directions*, pp. 7–53, 2017.

[27] N. P. Williams et al., "In vivo microelectrode arrays for neuroscience," *Nature Reviews Methods Primers*, vol. 5, no. 1, p. 31, 2025.

[28] E. N. Brown, R. E. Kass, and P. P. Mitra, "Multiple neural spike train data analysis: State-of-the-art and future challenges," *Nature neuroscience*, vol. 7, no. 5, pp. 456–461, 2004.

[29] G. Hong and C. M. Lieber, "Novel electrode technologies for neural recordings," *Nature Reviews Neuroscience*, vol. 20, no. 6, pp. 330–345, 2019.

[30] J. P. Seymour and D. R. Kipke, "Neural probe design for reduced tissue encapsulation in cns," *Biomaterials*, vol. 28, no. 25, pp. 3594–3607, 2007.

[31] L. Yang, K. Lee, J. Villagracia, and S. C. Masmanidis, "Open source silicon microprobes for high throughput neural recording," *Journal of neural engineering*, vol. 17, no. 1, p. 016 036, 2020.

[32] J. J. Jun et al., "Fully integrated silicon probes for high-density recording of neural activity," *Nature*, vol. 551, no. 7679, pp. 232–236, 2017.

[33] C. Grienberger, A. Giovannucci, W. Zeiger, and C. Portera-Cailliau, "Two-photon calcium imaging of neuronal activity," *Nature Reviews Methods Primers*, vol. 2, no. 1, p. 67, 2022.

[34] E. A. Pnevmatikakis, "Analysis pipelines for calcium imaging data," *Current opinion in neurobiology*, vol. 55, pp. 15–21, 2019.

[35] P. Gao et al., "A theory of multineuronal dimensionality, dynamics and measurement," *BioRxiv*, p. 214 262, 2017.

[36] J. A. Gallego, M. G. Perich, R. H. Chowdhury, S. A. Solla, and L. E. Miller, "Long-term stability of cortical population dynamics underlying consistent behavior," *Nature neuroscience*, vol. 23, no. 2, pp. 260–270, 2020.

[37] P. T. Sadtler et al., "Neural constraints on learning," *Nature*, vol. 512, no. 7515, pp. 423–426, 2014.

[38] C. Fefferman, S. Mitter, and H. Narayanan, "Testing the manifold hypothesis," *Journal of the American Mathematical Society*, vol. 29, no. 4, pp. 983–1049, 2016.

[39] R. Mitchell-Heggs, S. Prado, G. P. Gava, M. A. Go, and S. R. Schultz, "Neural manifold analysis of brain circuit dynamics in health and disease," *Journal of computational neuroscience*, vol. 51, no. 1, pp. 1–21, 2023.

[40] E. Scholz, *The concept of manifold, 1850–1950*. na, 1999.

[41] J. M. Shine et al., "Human cognition involves the dynamic integration of neural activity and neuromodulatory systems," *Nature neuroscience*, vol. 22, no. 2, pp. 289–296, 2019.

[42] H. Wang, Z. Lei, X. Zhang, B. Zhou, and J. Peng, "Machine learning basics," *Deep learning*, pp. 98–164, 2016.

[43] D. Bonvin, B. Srinivasan, and D. Hunkeler, "Control and optimization of batch processes," *IEEE Control Systems Magazine*, vol. 26, no. 6, pp. 34–45, 2006.

[44] L. Prechelt, "Early stopping-but when?" In *Neural Networks: Tricks of the trade*, Springer, 2002, pp. 55–69.

[45] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, 2020.

[46] C. Fortunato et al., "Nonlinear manifolds underlie neural population activity during behaviour," *BioRxiv*, pp. 2023–07, 2024.

[47] I. T. Joliffe and B. Morgan, "Principal component analysis and exploratory factor analysis," *Statistical methods in medical research*, vol. 1, no. 1, pp. 69–95, 1992.

[48] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[49] P. Reizinger et al., "Embrace the gap: Vaes perform independent mechanism analysis," *Advances in Neural Information Processing Systems*, vol. 35, pp. 12 040–12 057, 2022.

[50] S. Odaibo, "Tutorial: Deriving the standard variational autoencoder (vae) loss function," *arXiv preprint arXiv:1907.08956*, 2019.

[51] L. Pinheiro Cinelli, M. Araújo Marins, E. A. Barros da Silva, and S. Lima Netto, "Variational autoencoder," in *Variational methods for machine learning with applications to deep networks*, Springer, 2021, pp. 111–149.

[52] J. Shlens, "Notes on kullback-leibler divergence and likelihood," *arXiv preprint arXiv:1404.2000*, 2014.

[53] C. P. Burgess et al., "Understanding disentangling in

\

beta $-vae$," *arXiv preprint arXiv:1804.03599*, 2018.

[54] F. P. Casale, A. Dalca, L. Saglietti, J. Listgarten, and N. Fusi, "Gaussian process prior variational autoencoders," *Advances in neural information processing systems*, vol. 31, 2018.

[55] D. J. MacKay et al., "Introduction to gaussian processes," *NATO ASI series F computer and systems sciences*, vol. 168, pp. 133–166, 1998.

[56] R. Dey and F. M. Salem, "Gate-variants of gated recurrent unit (gru) neural networks," in *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*, IEEE, 2017, pp. 1597–1600.

[57] J. V. Stone, "Independent component analysis: An introduction," *Trends in cognitive sciences*, vol. 6, no. 2, pp. 59–64, 2002.

[58] A. Hyvarinen, H. Sasaki, and R. Turner, "Nonlinear ica using auxiliary variables and generalized contrastive learning," in *The 22nd international conference on artificial intelligence and statistics*, PMLR, 2019, pp. 859–868.

[59] V. Gallese, "The roots of empathy: The shared manifold hypothesis and the neural basis of intersubjectivity," *Psychopathology*, vol. 36, no. 4, pp. 171–180, 2003.

[60] C. M. Bishop, "Latent variable models," in *Learning in graphical models*, Springer, 1998, pp. 371–403.

[61] A. Dai, "Brain computer interface for the treatment of neurodegenerative diseases," in *E3S Web of Conferences*, EDP Sciences, vol. 553, 2024, p. 05 010.

[62] B. Fiani et al., "An examination of prospective uses and future directions of neuralink: The brain-machine interface," *Cureus*, vol. 13, no. 3, 2021.

[63] J. J. Shih, D. J. Krusienski, and J. R. Wolpaw, "Brain-computer interfaces in medicine," in *Mayo clinic proceedings*, Elsevier, vol. 87, 2012, pp. 268–279.

[64] E. Y. Wang et al., "Foundation model of neural activity predicts response to new stimulus types," *Nature*, vol. 640, no. 8058, pp. 470–477, 2025.

[65] A. D. Grosmark and G. Buzsáki, "Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences," *Science*, vol. 351, no. 6280, pp. 1440–1443, 2016.

[66] D. Zhou and X.-X. Wei, "Learning identifiable and interpretable latent models of high-dimensional neural activity using pi-vae," *Advances in neural information processing systems*, vol. 33, pp. 7234–7247, 2020.

[67] R. H. Chowdhury, J. I. Glaser, and L. E. Miller, "Area 2 of primary somatosensory cortex encodes kinematics of the whole arm," *Elife*, vol. 9, e48198, 2020.

[68] T. M. Ball, L. M. Squeglia, S. F. Tapert, and M. P. Paulus, "Double dipping in machine learning: Problems and solutions," *Biological Psychiatry: Cognitive Neuroscience and Neuroimaging*, vol. 5, no. 3, pp. 261–263, 2020.

[69] A. Di Bucchianico, "Coefficient of determination (r 2)," *Encyclopedia of statistics in quality and reliability*, 2008.

[70] J. Cornell and R. Berger, "Factors that influence the value of the coefficient of determination in simple linear and nonlinear regression models," *Phytopathology*, vol. 77, no. 1, pp. 63–70, 1987.

[71] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, "Linear regression," in *An introduction to statistical learning: With applications in python*, Springer, 2023, pp. 69–134.

[72] S. Ramraj, N. Uzir, R. Sunil, and S. Banerjee, "Experimenting xgboost algorithm for prediction and classification of different datasets," *International Journal of Control Theory and Applications*, vol. 9, no. 40, pp. 651–662, 2016.

[73] S. Kang, K. Li, and R. Wang, "A survey on pareto front learning for multi-objective optimization: S. kang et al.," *Journal of Membrane Computing*, vol. 7, no. 2, pp. 128–134, 2025.

[74] A. Díaz Lantada et al., "Cdio experiences in biomedical engineering: Preparing spanish students for the future of medicine and medical device technology," 2016.

[75] K. N. K. C. Sunkara, "Power consumption and heat dissipation in ai data centers: A comparative analysis," 2025.

[76] P. Aguilera-Benito and M. Baquero-Arenal, "Assessment of renewable energy integration and future development of spain's electrical system in alignment with eu policy," *Discover Energy*, 2026.

[77] E. Maestri, "The 3rs principle in animal experimentation: A legal review of the state of the art in europe and the case in italy," *BioTech*, vol. 10, no. 2, p. 9, 2021.

[78] A. W. Act, *Animal welfare act*, 2006.

[79] N. I. of Health (US). Office for Protection from Research Risks, *Public Health Service policy on humane care and use of laboratory animals*. Office for Protection from Research Risks (OPRR), National Institutes of Health, 1986.

[80] I. A. Care, "Institutional animal care and use committee,"

[81] M. Mahler, C. Auza, R. Albesa, C. Melus, and J. A. Wu, "Regulatory aspects of artificial intelligence and machine learning-enabled software as medical devices (samd)," in *Precision medicine and artificial intelligence*, Elsevier, 2021, pp. 237–265.

[82] L. Svempe, "Exploring impediments imposed by the medical device regulation eu 2017/745 on software as a medical device," *JMIR Medical Informatics*, vol. 12, no. 1, e58080, 2024.

[83] J. Black and A. D. Murray, "Regulating ai and machine learning: Setting the regulatory agenda," *European journal of law and technology*, vol. 10, no. 3, 2019.

[84] M. Ienca, G. Valle, and S. Raspopovic, "Clinical trials for implantable neural prostheses: Understanding the ethical and technical requirements," *The Lancet Digital Health*, vol. 7, no. 3, e216–e224, 2025.

A Appendix A

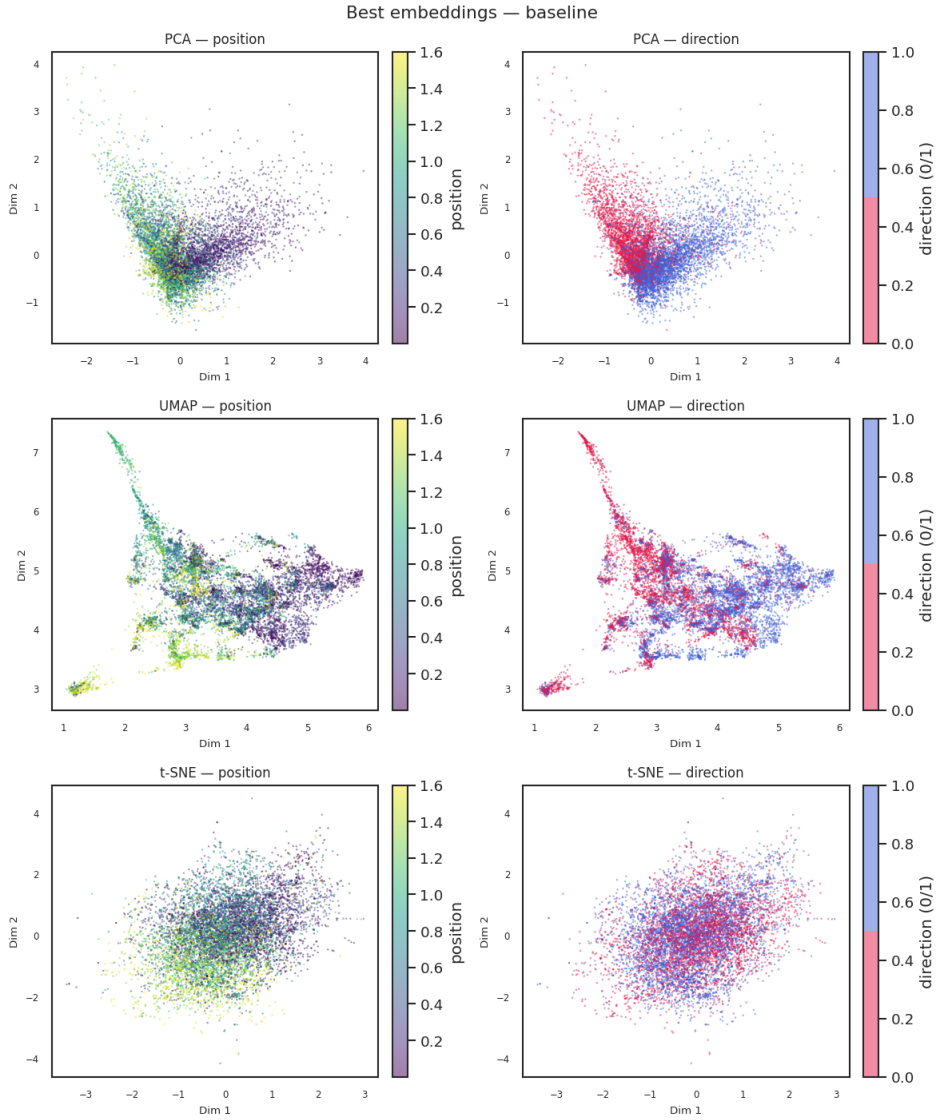

Figure A.1: Embeddings of the baseline LVMs on the rat hippocampus dataset.

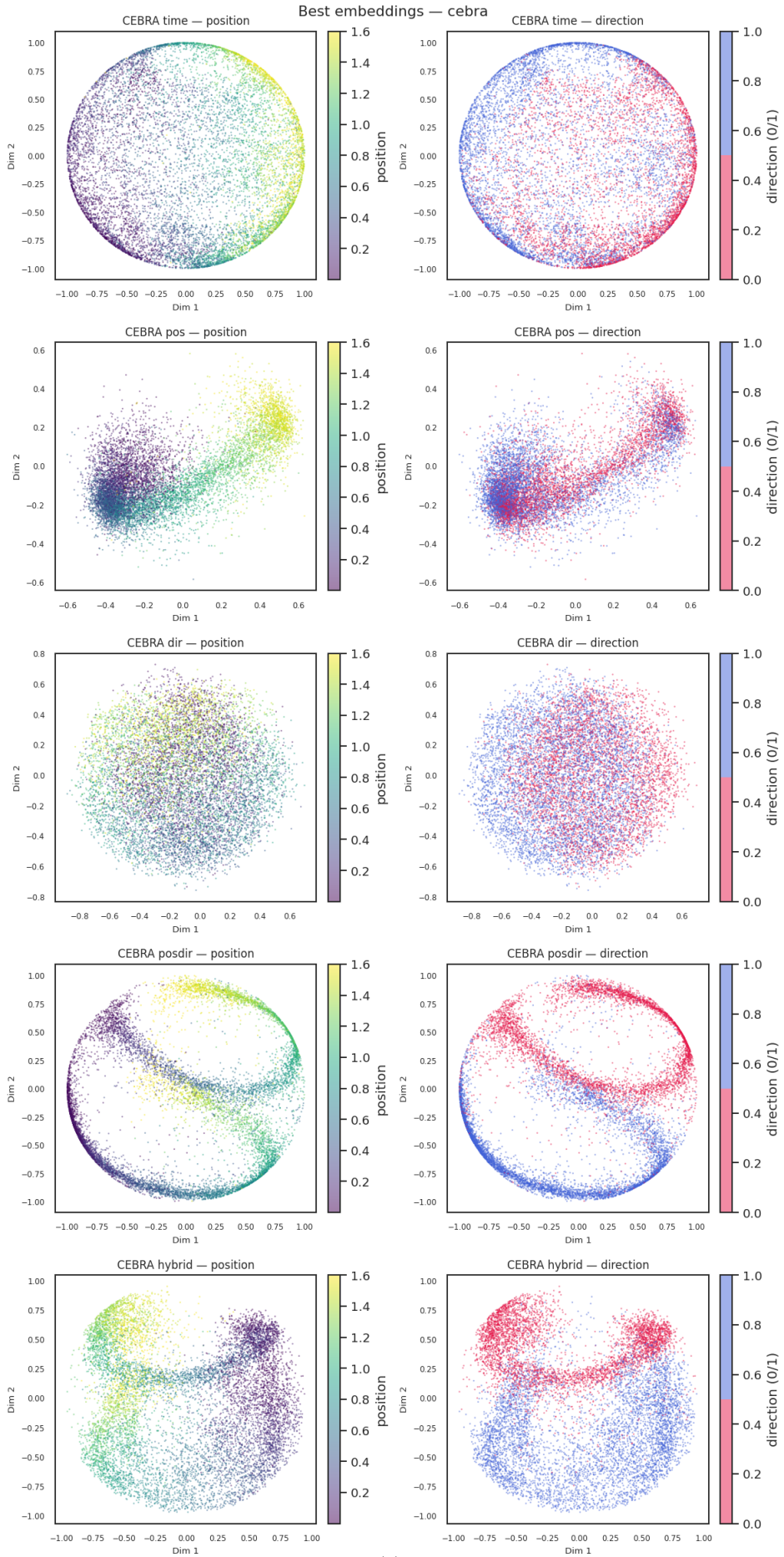

Figure A.2: Embeddings of CEBRA on the rat hippocampus dataset.

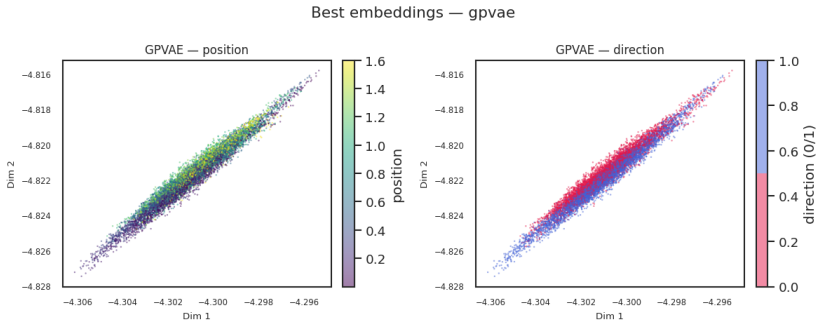

Figure A.3: Embeddings of GP-VAE on the rat hippocampus dataset.

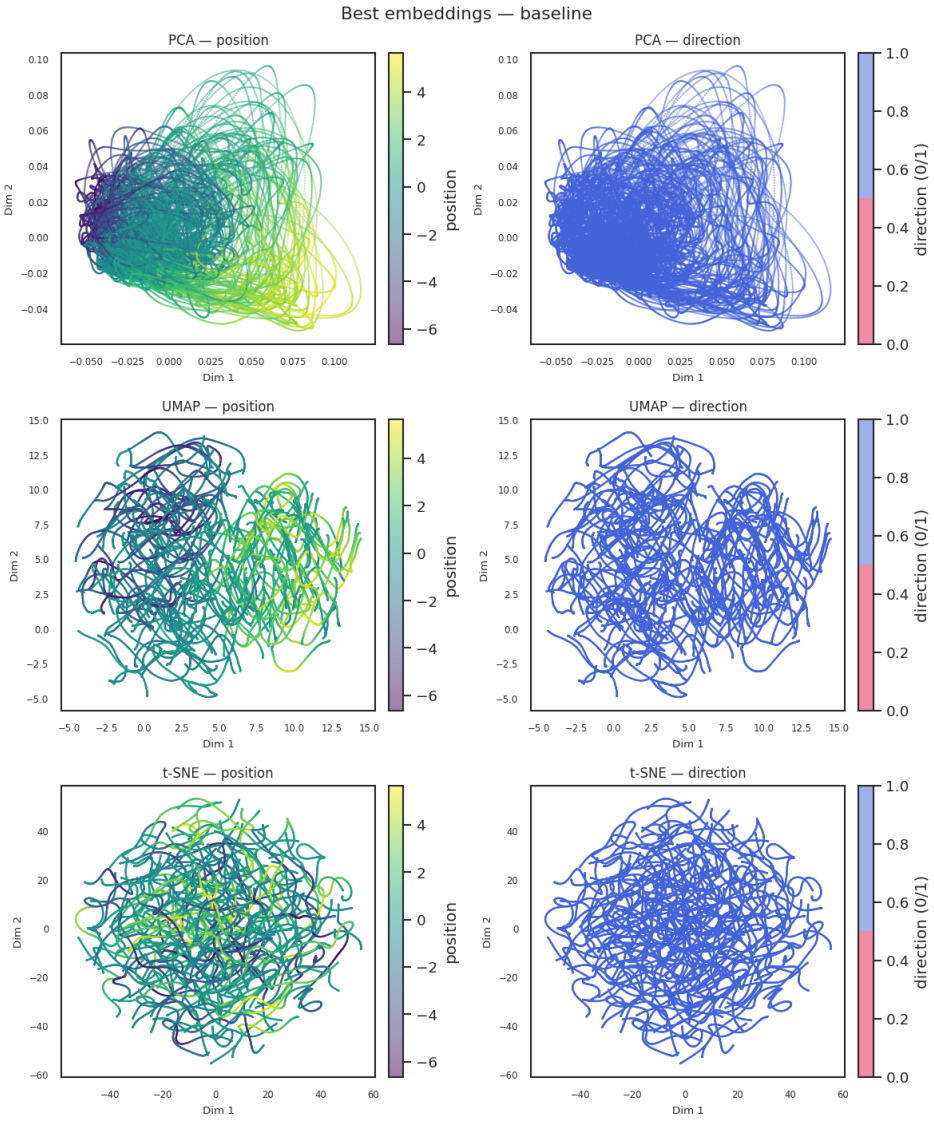

Figure A.4: Embeddings of the baseline LVMs on the macaque reaching dataset.

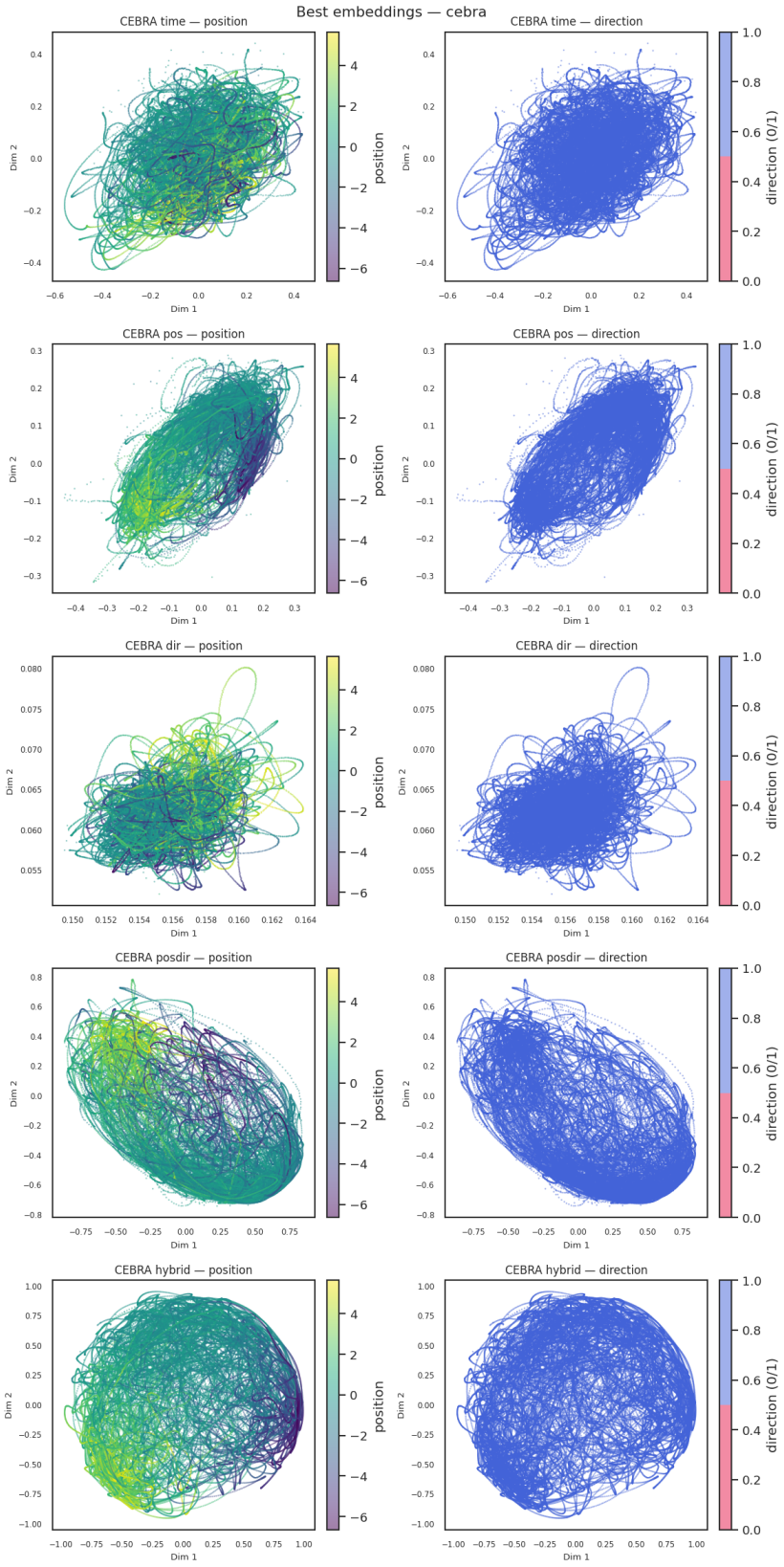

Figure A.5: Embeddings of CEBRA on the macaque reaching dataset.

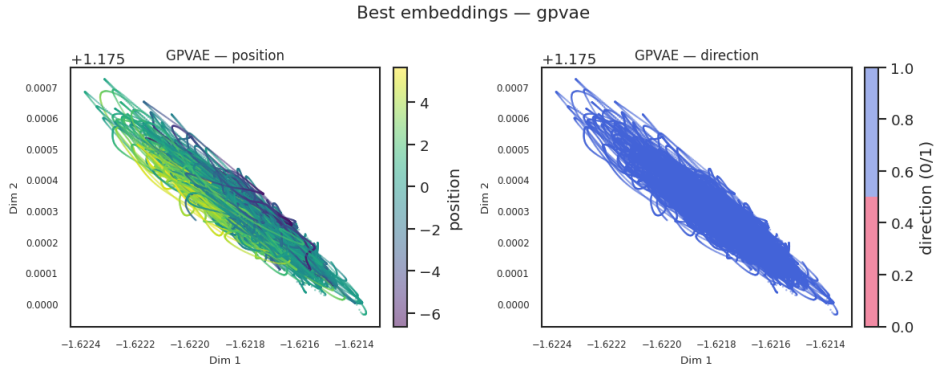

Figure A.6: Embeddings of GP-VAE on the macaque reaching dataset.

B Appendix B

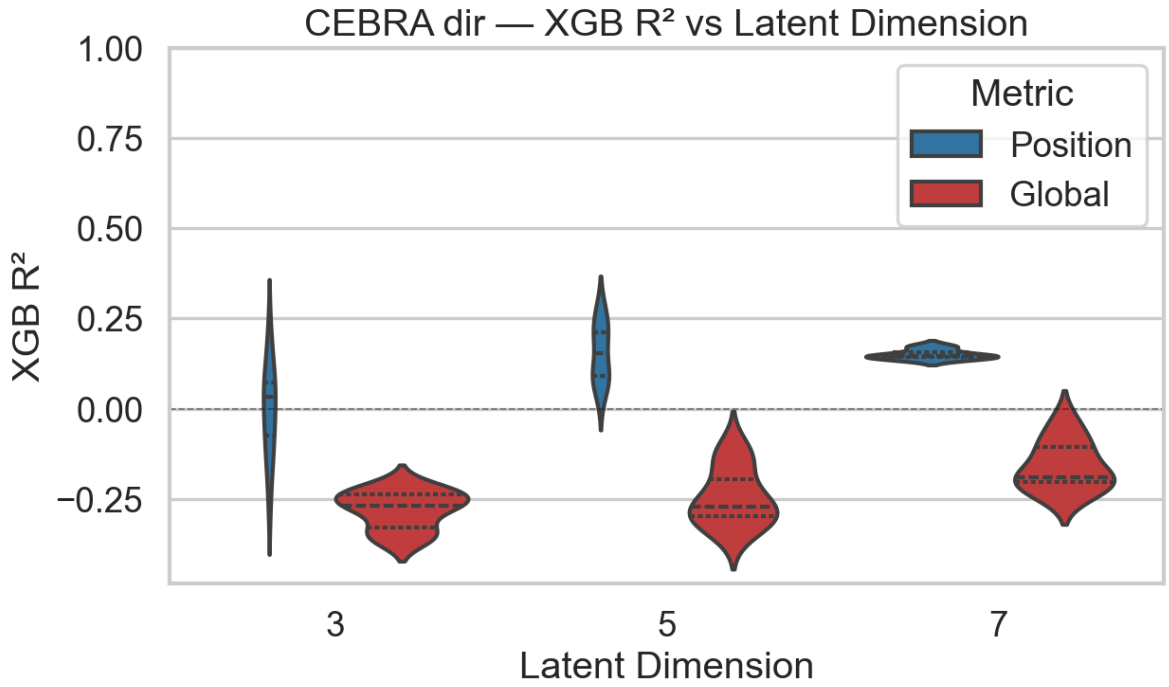

Figure B.1: Violin plot of CEBRA-Dir across latent dimensions in the rat hippocampus benchmark.

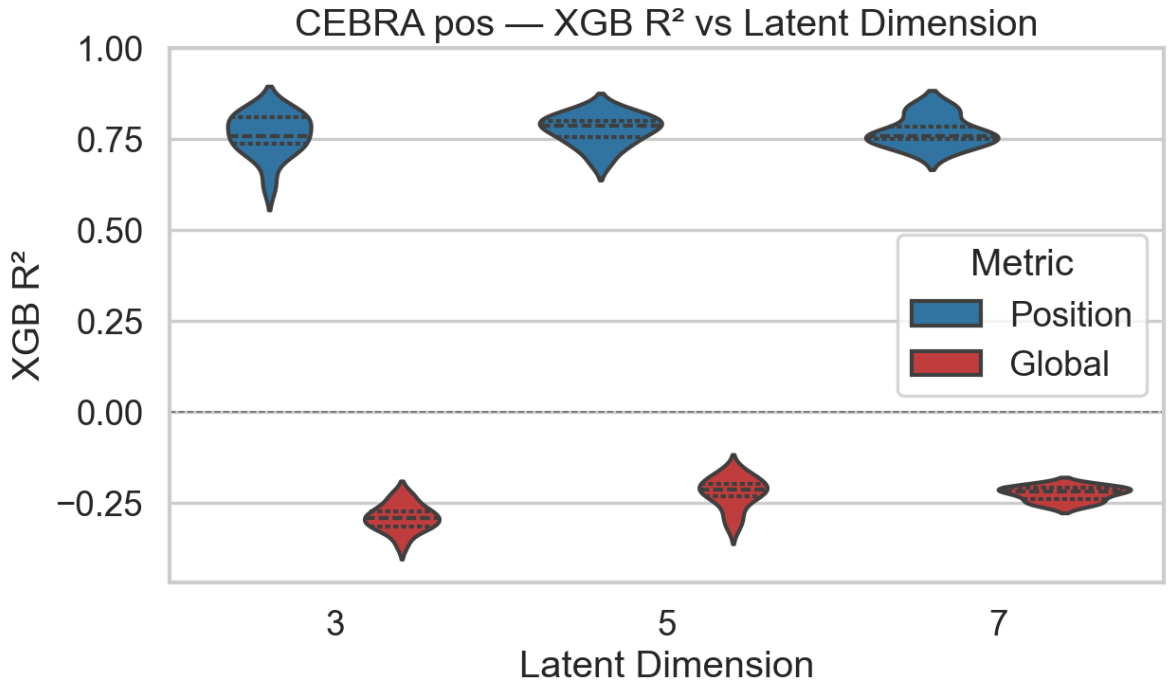

Figure B.2: Violin plot of CEBRA-Pos across latent dimensions in the rat hippocampus benchmark.

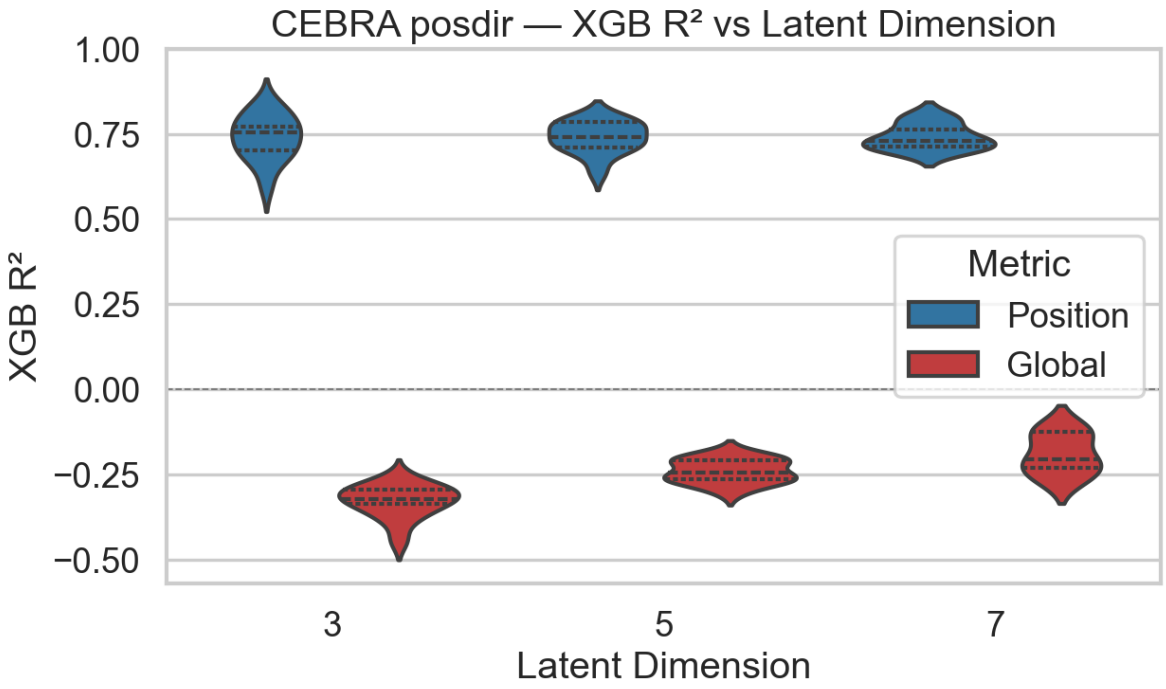

Figure B.3: Violin plot of CEBRA-PosDir across latent dimensions in the rat hippocampus benchmark.

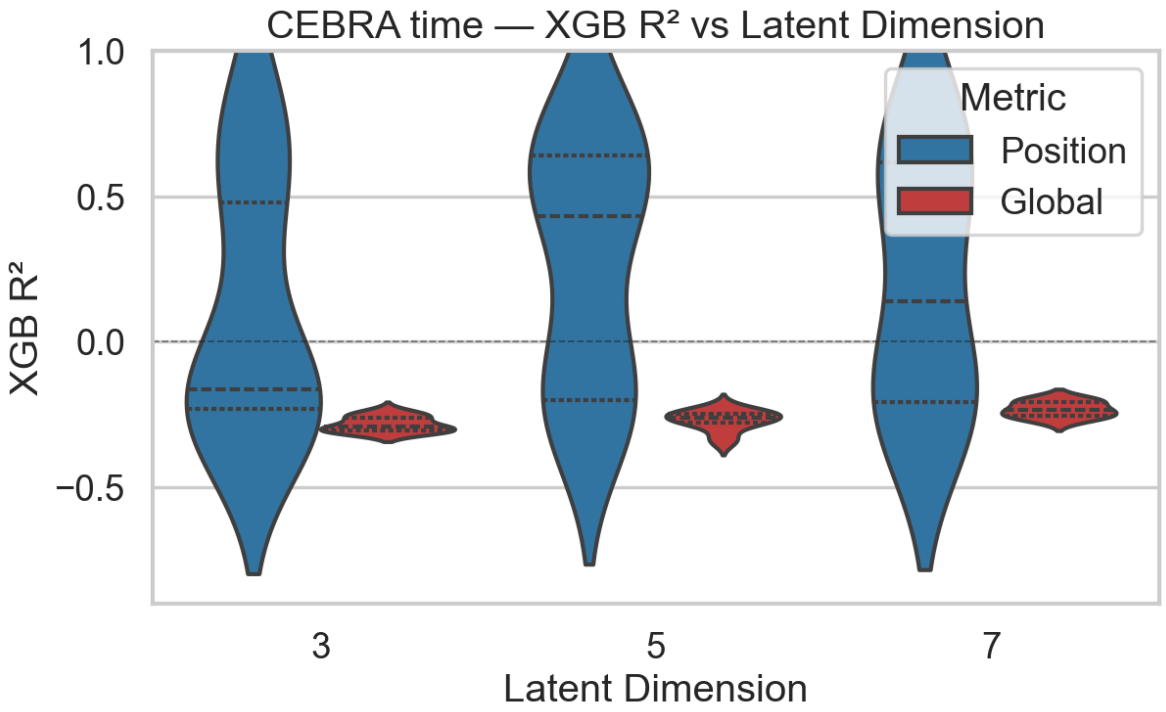

Figure B.4: Violin plot of CEBRA-Time across latent dimensions in the rat hippocampus benchmark.

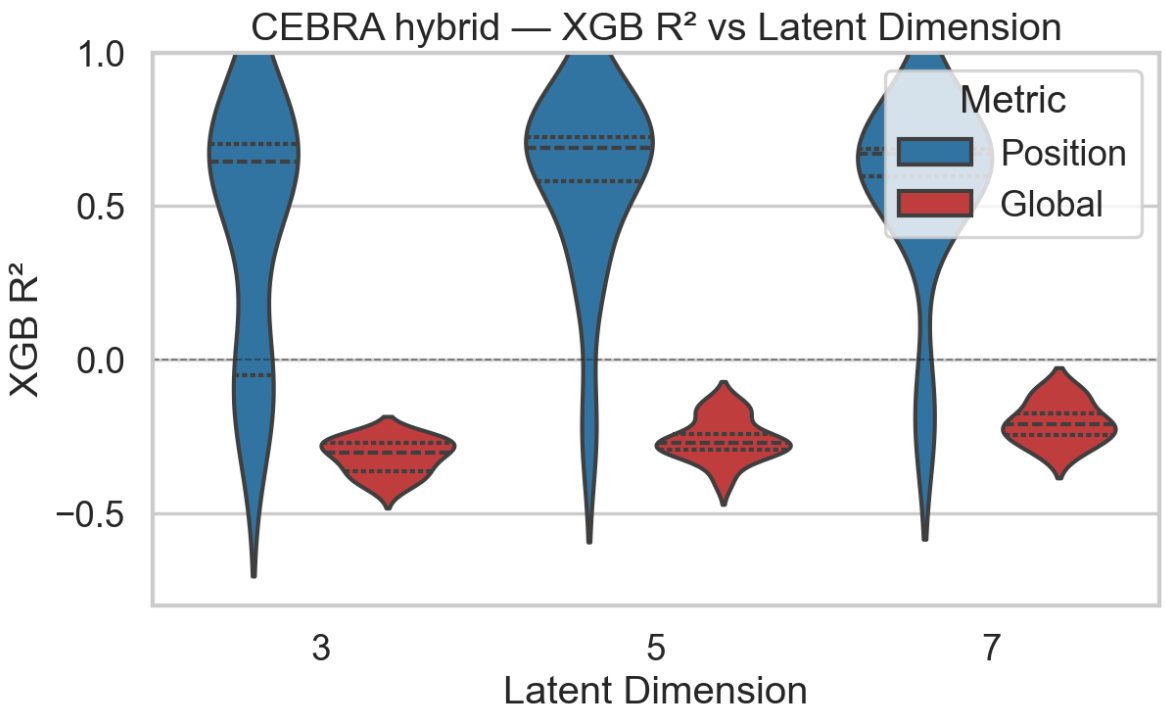

Figure B.5: Violin plot of CEBRA-Hybrid across latent dimensions in the rat hippocampus benchmark.

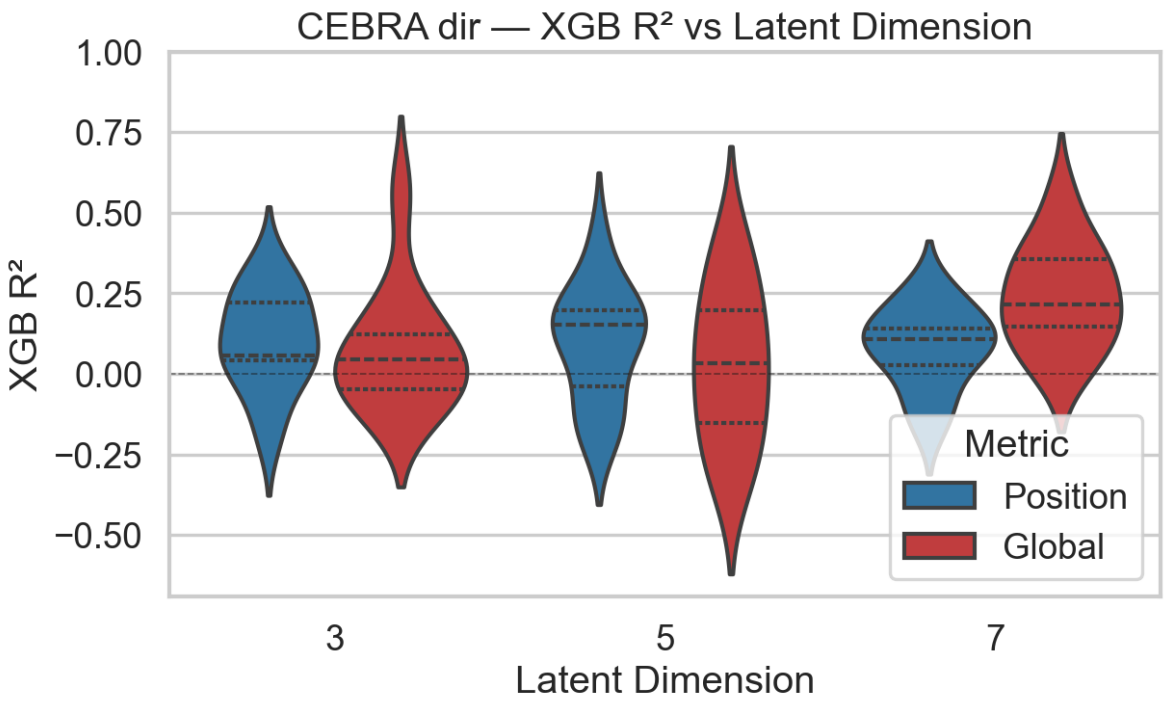

Figure B.6: Violin plot of CEBRA-Dir across latent dimensions in the macaque reaching dataset.

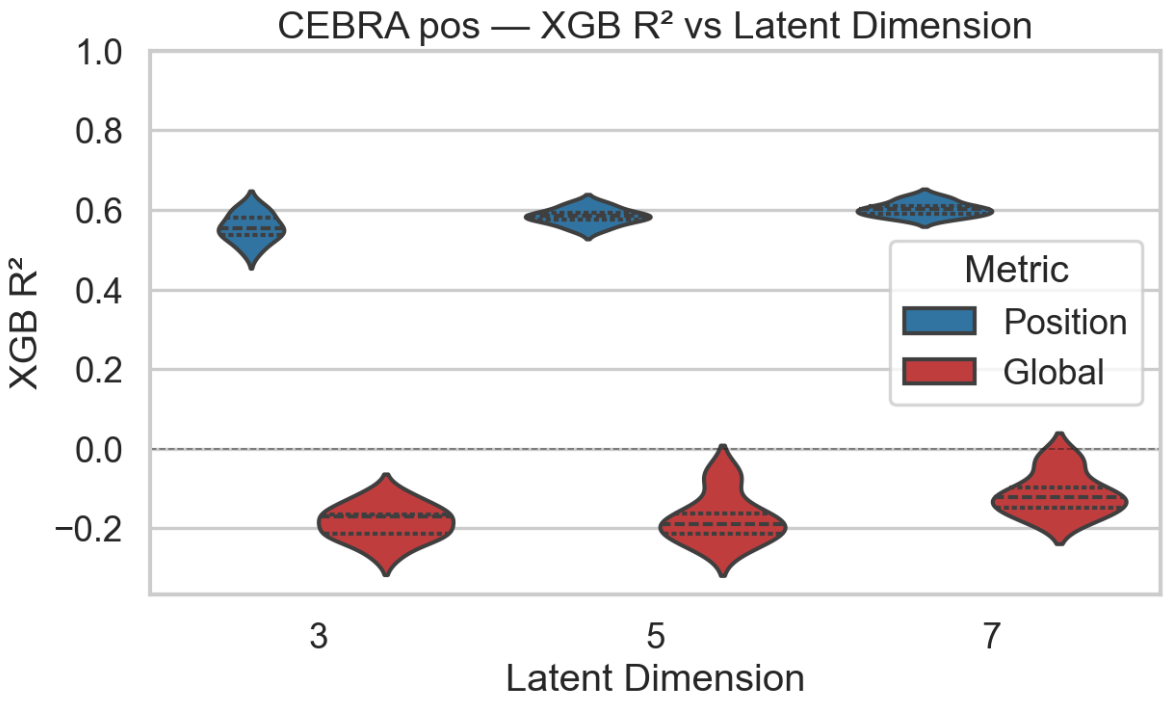

Figure B.7: Violin plot of CEBRA-Pos across latent dimensions in the macaque reaching dataset.

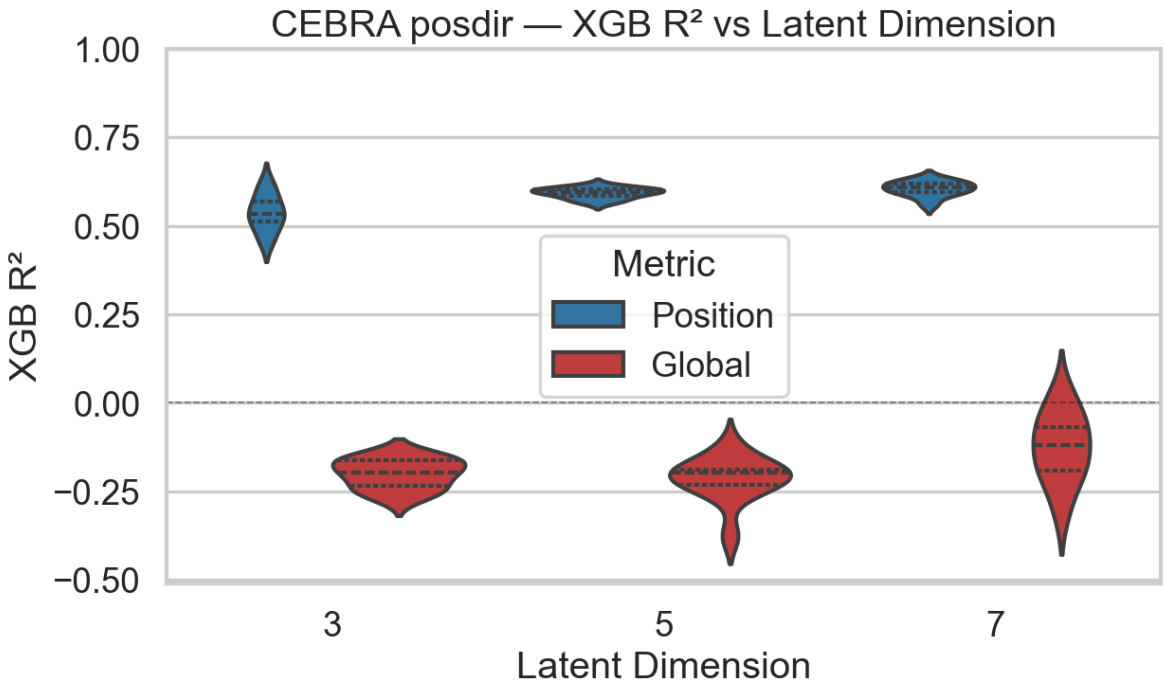

Figure B.8: Violin plot of CEBRA-PosDir across latent dimensions in the macaque reaching dataset.

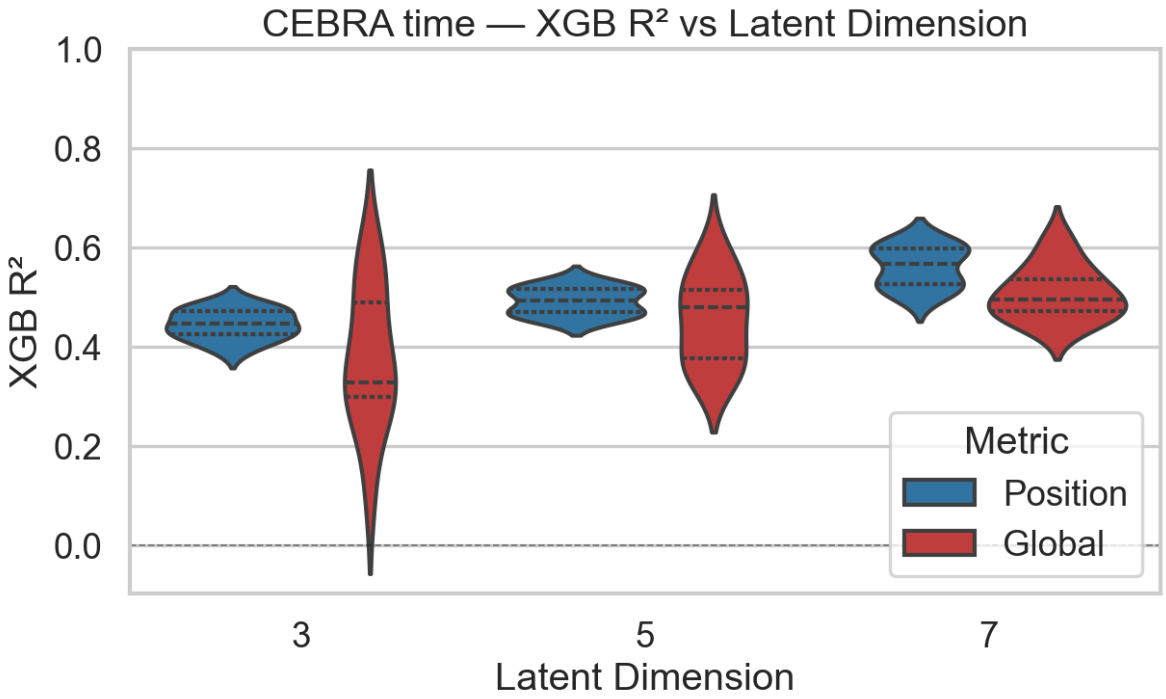

Figure B.9: Violin plot of CEBRA-Time across latent dimensions in the macaque reaching dataset.

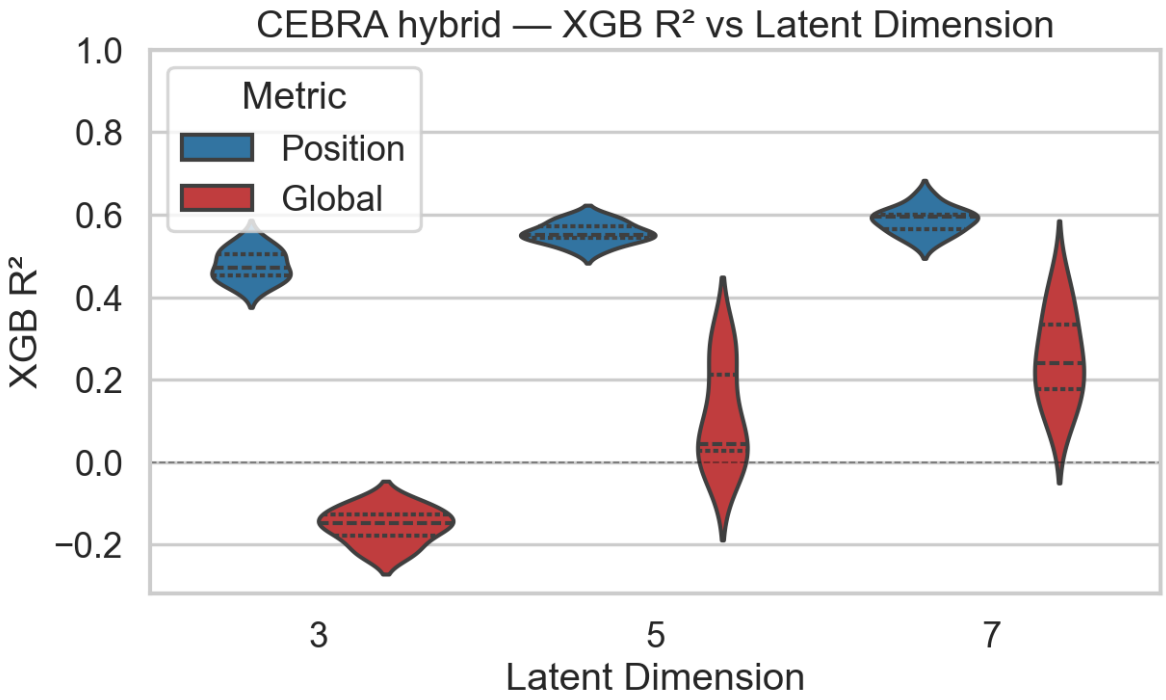

Figure B.10: Violin plot of CEBRA-Hybrid across latent dimensions in the macaque reaching dataset.

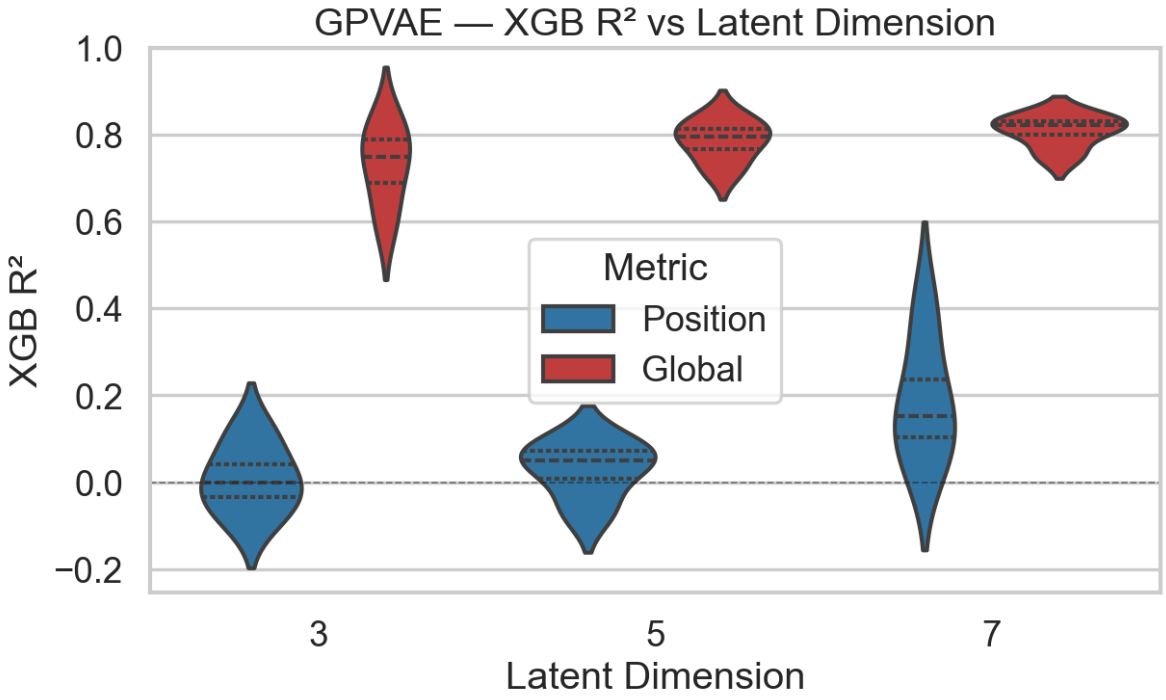

Figure B.11: Violin plot of GP-VAE across latent dimensions in the macaque reaching dataset.

C Appendix C

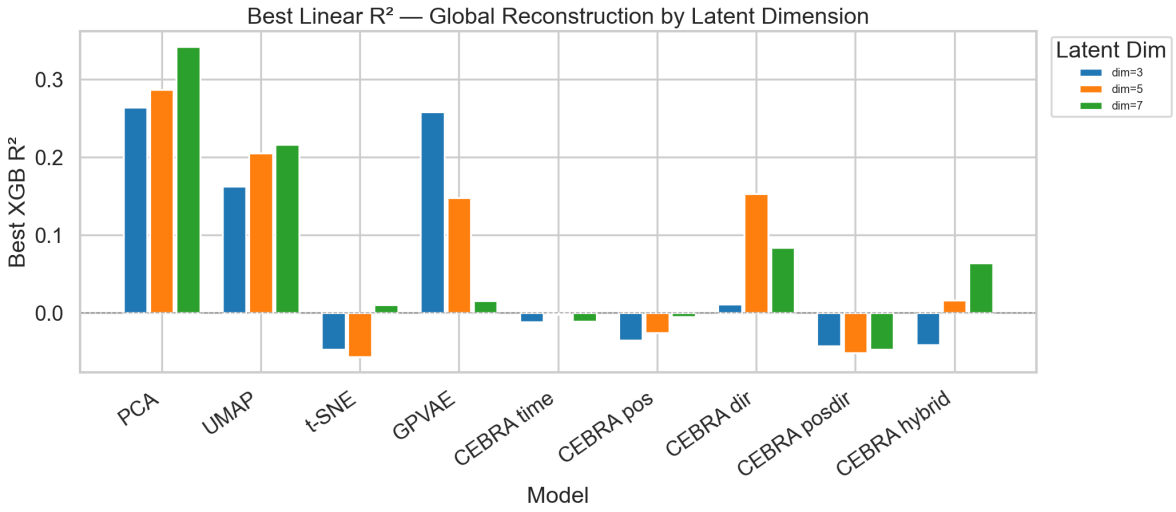

Figure C.1: Barplot of the best R^2 scores for global activity reconstruction of each model variety using linear estimators over latent dimensions for rat hippocampus dataset

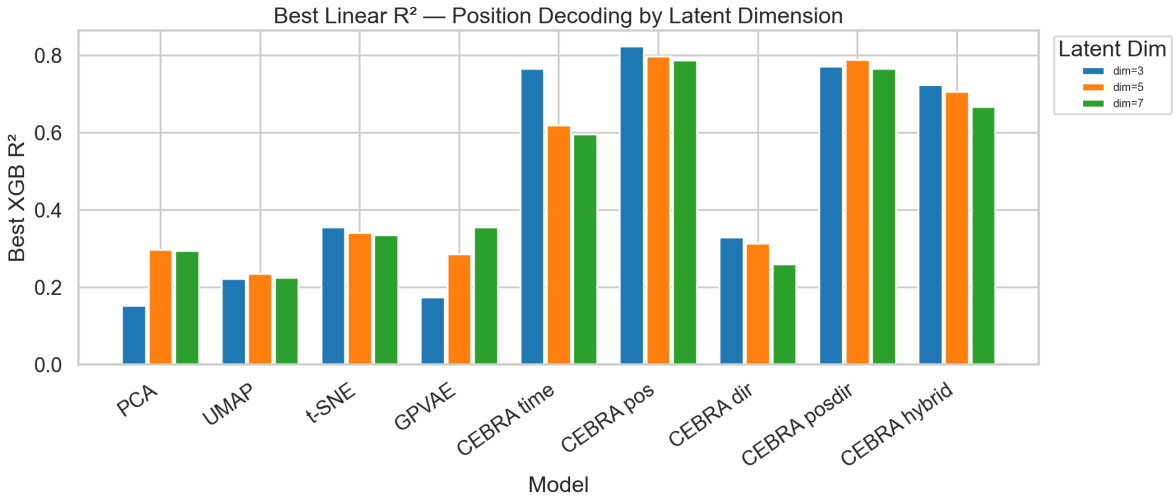

Figure C.2: Barplot of the best R^2 scores for positional decoding of each model variety using linear estimators over latent dimensions for rat hippocampus dataset

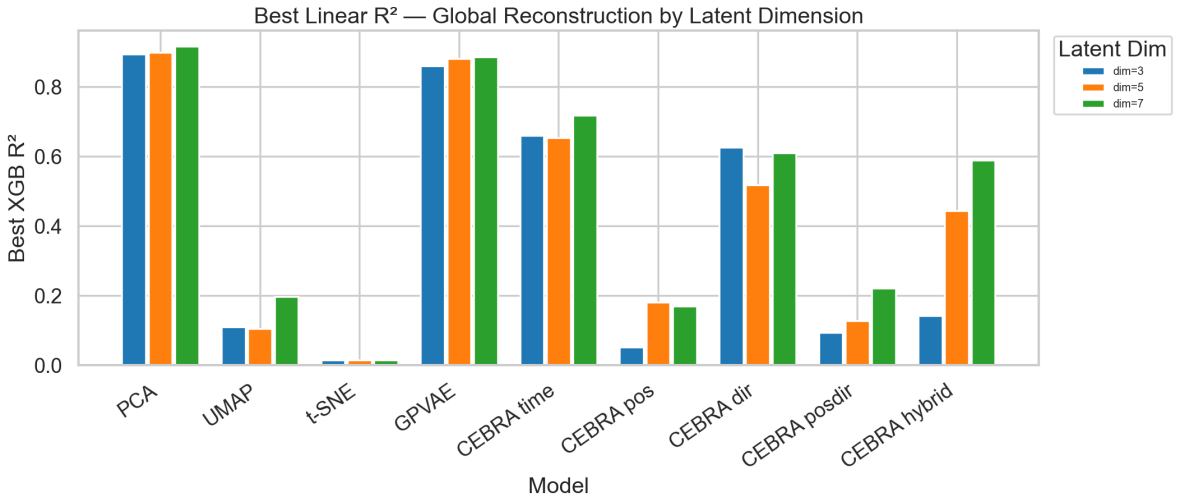

Figure C.3: Barplot of the best R^2 scores for global activity reconstruction of each model variety using linear estimators over latent dimensions for macaque reaching dataset

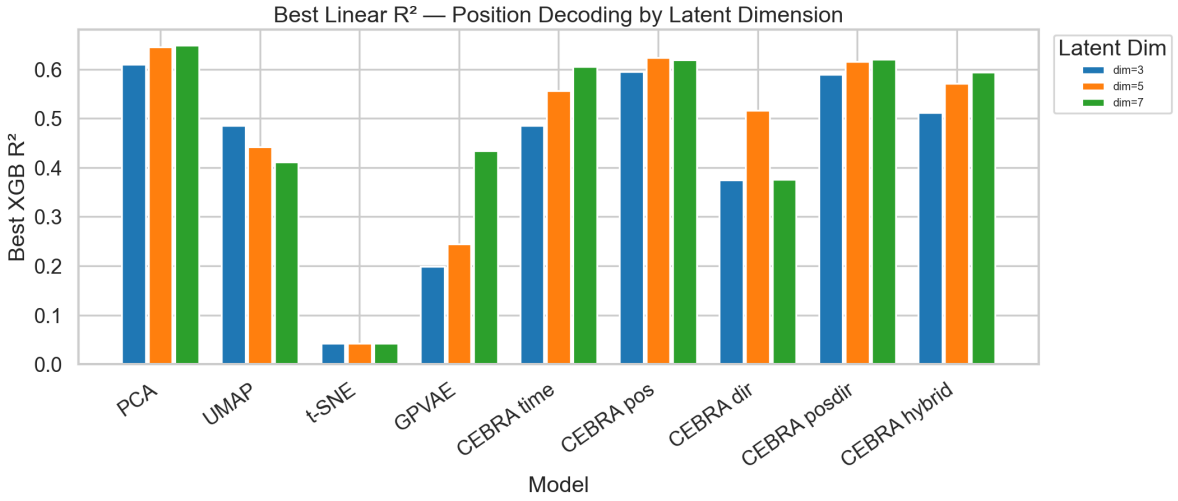

Figure C.4: Barplot of the best R^2 scores for positional decoding of each model variety using linear estimators over latent dimensions for macaque reaching dataset