Enabling naturalistic, long-duration and continual animal experimentation

March 25, 2025

Contents

1 Vision

For over four years, at the Sainsbury Wellcome Centre and Gatsby Computational Neuroscience Unit, we have been developing the AEON platform, a set of hardware and software tools that support a new type of experimentation, where animals are allowed to express ethologically-relevant behaviours, in naturalistic environments, and in long-duration experiments, while their behaviour and neural activity is monitored continuously for weeks to months. We have used this platform to characterize foraging behaviour in both solitary and groups of mice (**?**) (Figure **??**).

Our US partner, the Allen Institute for Neural Dynamics, is using the AEON platform in continuous learning experiments, where mice freely explore odors continuously for days to weeks (**?**).

This is an unprecedented type of experimentation that ...

Several groups around the world are performing this new type of experimentation .

We have built the AEON platform, and have used it to collect weeks- to months-long NaLoDuCo experimental data. We next propose to develop advanced machine learning methods and intelligent visualisations to extract meaning from this data (Aim 1).

A central aim of both the SWC/GCNU and AIND is to contribute to open science. We thus propose to create software infrastructure to openly disseminate NaLoDuCo recordings, visualisation and data analysis methods (Aim 2).

Over more than four years we have developed the AEON platform following high software engineering practices. It is an open source platform that anybody can use and modify (**?**). We want it to become the standard platform for the collection of NaLoDuCo experimental data. We are currently using AEON on two new NaLoDuCo experiments: (1) odor learning experiments, lasting for days to weeks, lead by Dr. Carl Schoonover at the AIND, and (2) foraging experiments in very large arenas (eight meters in diameter), lead by Prof. Tiago Branco at the SWC. We will extend and validate the functionality of the AEON platform by applying it to these and new NaLoDuCo experiments. A key functionality that we propose to add as part of this project is real-time machine learning, to allow to control AEON experiments with live inferences (Aim 3).

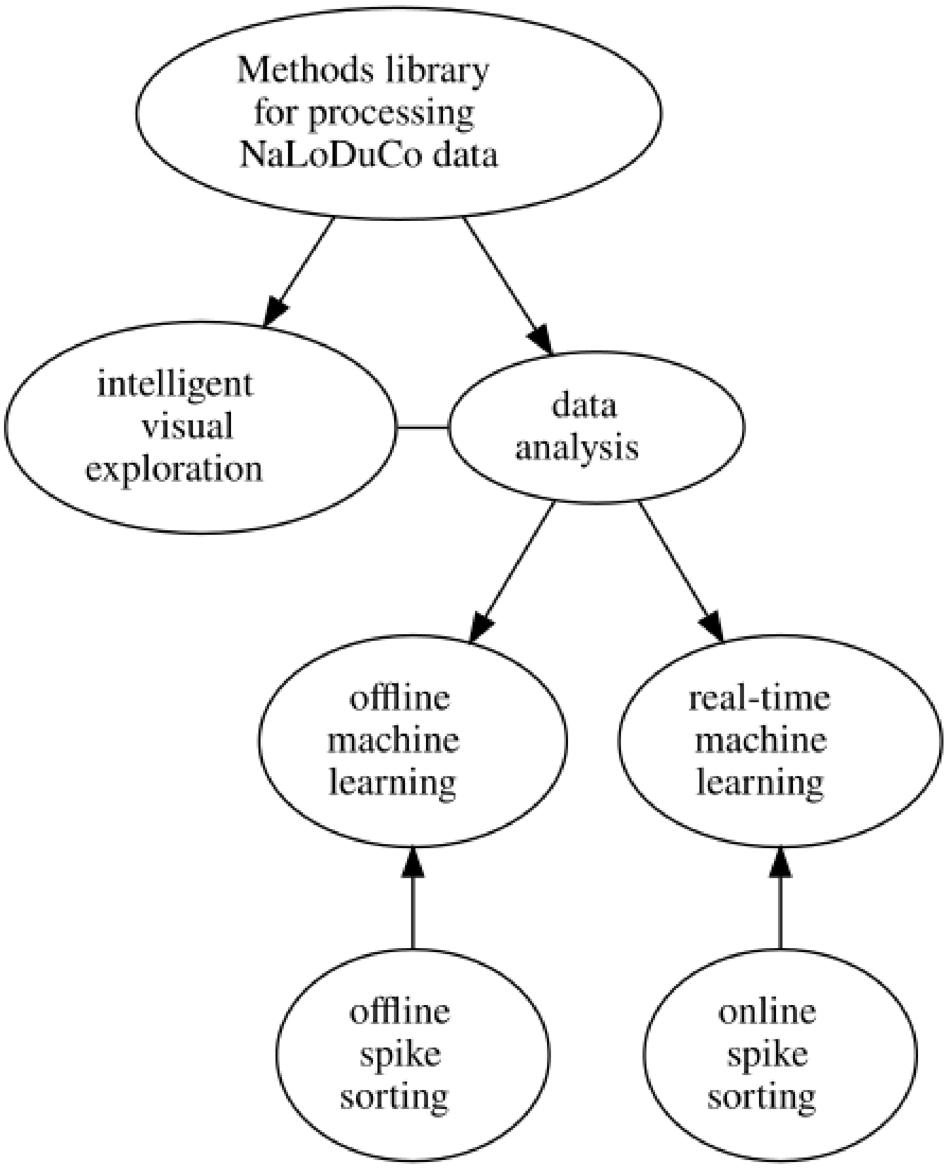

Figure 1: Proposal aims

Aim 1: create infrastructure for open dissemination of NaLoDuCo experimental recordings

The dissemination of NaLoDuCo recordings is not trivial, as datasets generated by this new type of experimentation are enormous. For instance, the size of a dataset generated from a one week recording of behavioural and neural activity from a foraging mouse in SWC experiments exceeds 200 terabytes. It will take users several days to download these datasets over standard Internet connections.

Instead of bringing data to users, we will bring users to data, by storing datasets in the cloud (or in institutional clusters), and providing **cloud software to allow users to visually explore and statistically analyse behavioural and neural NaLoDuCo datasets where they live** (Figure 1, left box).

Our statistical analysis of neural time series will require knowledge of the spiking activity of single units; i.e., spike sorting. In long-duration experiments with freely moving animals spike sorting is a challenging problem, because movements of recording probes change the shape of spike waveforms over time and complicate the assignment of spikes to units based on their waveforms. We will address this problem by developing **spike sorting methods for long-duration and continual, long-duration and high-channel-count recordings** (Figure 1, left box).

Aim 2: create real-time machine learning methods for intelligent experimentation

In small-animal Neuroscience, most often statistical processing of neural time series is performed offline; i.e., experimental data is collected, saved to files, which are later statistically processed, with no runtime constraints. Most often all experimental data is processes at the same time; i.e., batch processing.

A new online statistical processing approach is now emerging in small-animal Neuroscience, where data is processes while it is being collected, and at the speed of data generation (**?**).

Online methods are well suited for NaLoDuCo experimentation. In experiments extending for weeks to months animals learn and adapt, their motivation and fatigue may fluctuate, and experimental conditions (e.g., lighting) may change. Offline batch processing algorithms cannot model this type of changing data. They assume stationary data whose statistical properties do not change across time. Differently, most online processing algorithms are robust to these changes. Also, NaLoDuCo experimentation is well suited for online methods, as the long-duration of these experiments provide a large amount of data to accurately fit expressive online methods.

We will **optimize methods developed for Aim 1 so that they can operate in real time**, and focus on the following two applications of these online methods (Figure 1, right box).

Intelligent neuromodulation

Brain activity can be modulated optically, chemically and electrically (). Most commonly this modulations is done at fixed experimental times, or based on simple behavioural or neural observations.

We will guide optogenetic manipulations based on inferences from advanced machine learning methods. For example, a scientists may hypothesize that a peak in a neural latent variable, inferred from a prefrontal cortex population, signals the moment when mice decide to begin a foraging bout. To test this, she runs an online machine learning model to estimate latent variables from prefrontal cortex activity, predicting when this peak will occur. She then optogenetically inactivates the neural population at the forecasted time. Because inactivation prevented the mouse from initiating a foraging bout, her hypothesis is supported.

Intelligent experimental data storage

As the duration of NaLoDuCo experiments become longer, and the richness of the behavioural and neural recordings become larger, it will be unfeasible to store all raw data. We will be forced to intelligently decide, in real time, subsets of data to discard.

For instance, if we are recording videos from a mouse foraging in a large arena with ten high-resolution cameras, it would save considerable storage if at any time we only save videos from cameras capturing the mouse at that time. This could be done by tracking the position of the mouse in real time with probabilistic machine learning methods. Then, when the confidence of the tracking is high, we would only save videos of cameras capturing the mouse at the tracked position, but when the confidence is low, we would save all videos.

2 Approach

We have collected unprecedented NaLoDuCo datasets at the SWC. However, these very large datasets are of not much use without methods to visualise and analyse them. Sections 2.1 and 2.2 present such methods.

2.1 Offline analysis methods

We will disseminate an open-source library of advanced statistical methods adapted to **efficiently** process **non-stationary** recordings from NaLoDuCo experiments. We will benchmark the performance of these methods for processing behavioural and neural time series recorded from the SWC NaLoDuCo foraging experiments.

The initial list of methods to include in this library is given in Section 2.1.1 and includes regression, classification, supervised, unsupervised and reinforcement learning, state space models, artificial neural networks and transformers. Implementations of these methods will follow high software engineering standards, and will include detailed documentation, so that users can easily apply them to process their own NaLoDuCo datasets. This library should be built for all and by all, and become essential to the rapidly expanding community of scientists performing NaLoDuCo experimentation in small animals around the world ().

In Neuroscience we don't have methods to characterise long-duration and continual time series, to learn from time series whose statistical properties fluctuate over time, to forecast time series over long horizons (e.g., hours, days, week or months). This library should find applicability beyond the realm of small-animal neuroscience and, for example, be useful to characterize long-duration and continual neural processes in human brain activity measured with subscalp EEG electrodes ().

2.1.1 Initial list of methods to include in the library

The first step in the analysis of NaLoDuCo foraging behavioural data is **tracking multiple body parts** in mice. For this we will use **deep learning** methods, as in (**?**). Next, we will use the previous tracking outputs to **infer mice kinematics** with **linear dynamical models**, as in (**?**). We will combine the tracking outputs with the kinematics inferences to **infer behavioural states** with **hidden Markov models**, as in (**?**). Further, we will **related kinematics and behavioural states to the probability of foraging events**, like leaving a patch, with **generalized linear models**, as in (), and **artificial neural networks**, as in (). The final step of the behavioural analysis will be to **infer mice policy from behavioural measures** with **inverse reinforcement learning**, as in **?**.

The characterization of neural data will begin with the **estimation of latent variables models**, to reduced the dimensionality of multielectrode recordings of hundreds or even thousands of neurons to a small number of latent variables, using **latent variable models**, with linear (**?**) and non-linear (**?**) latent dynamics.. We will use the estimated latent variables as inputs to **infer neural states**, using **HMMs**, as in (). Next, we will **decode mice position** from hippocampal recordings, and **study replay** during long-duration **foraging**, with **point process decoders**, as in (**?**).

Domain	**Functionality**	**Method**	**Model**
behaviour	multi-body-part tracking	SLEAP	deep neural network
behaviour	kinematics inference	lds_python	linear dynamical system
behaviour	kinematics inference	lds_python	particle filter
behaviour	behavioural state inference	SSM	hidden Markov model
behaviour	behavioural predictions	???	generalized linear model
behaviour	behavioural predictions	???	deep neural network
behaviour	policy inference	L(M)V-IQL	reinforcement learning
behaviour	forecasting behaviour	???	RNN
behaviour	forecasting behaviour	???	transformers
brain	latents inferences	svGPFA	Gaussian processes
brain	latents inferences	LFADS	RNN

brain	neural state inferences	SSM	hidden Markov model
brain	decoding	NA	point-process decoder
brain & behaviour	latents inference	RPM	Bayesian inference + deep neural network
brain & behaviour	latents inference	CEBRA	contrastive learning

2.1.2 Challenges

Extracting meaning from long-duration and continual recordings opens challenges and opportunities that we will address and exploit in this project, as we describe in this and the next sections.

Non-stationarity

Conventional offline methods used to characterize neural time series assume that the statistical characteristics of the modeled data do not change with time (i.e., that the probability of the data is time invariant – stationarity). This assumption may be valid for shorter experiments. However, for long-duration experiments, where animals learn and adapt, where their motivation fluctuates, and their activity is modulated by circadian, utradiem and peridiem rhythms, this assumption may not hold. In nonstationary environments, a non-adaptive model trained under the false stationarity assumption is bound to become obsolete in time, and perform sub-optimally at best, or fail catastrophically at worst. Below we briefly describe the type of methods we will use to adapt the disseminated methods to non-stationary environments.

The field of adaptive signal processing develops algorithms to characterize non-stationary systems (**?**). In this field adaptations to specific algorithms have been developed to improve their performance in non-stationary environments.

For example, the recursive least-squares algorithm (**?**, Chapter 9) is an adaptation of the ordinary least square algorithm to perform **linear regression** with non-stationary data.

For non-linear regression using **artificial neural networks**, a very large number of strategies have been developed to address data non-stationarity.

To mention a few, continual learning has introduced algorithms like Elastic Weight Consolidation (EWC) and Learning Without Forgetting (LwF) to allow models to adapt to changes over time without catastrophic forgetting. Also from this subfield is the Experience Replay (ER) algorithm that stores past data samples in a buffer and replays them alongside new data during training. A different type of strategy is used by ensemble methods (), which combine multiple models trained on different time windows to capture evolving data patterns.

Algorithms for **state-space models**, such as the Kalman filter, perform well in relatively simple non-stationary environments where data exhibit a Gaussian distribution with time-varying mean and covariance. However, in more complex settings with abrupt regime shifts or structured variability, more flexible approaches are required.

Switching state-space models, such as the switching linear dynamical system (SLDS) and the switching Hidden Markov model (sHMM), address discrete changes in system dynamics by adapting to different latent states. For tracking nonlinear and non-Gaussian processes, particle filters offer a powerful alternative by approximating posterior distributions through sequential sampling. Additionally, Bayesian online learning provides a principled framework for adapting probabilistic models to evolving data distributions, enabling continual adaptation in dynamic environments.

In the machine learning literature the study of non-stationary systems is done under the label of **concept drift** (), which refers to a change in the statistical properties of data that causes a model to perform poorly. Differently from adaptive signal processing, most methods developed to tackle concept drift are model agnostic and can be used with multiple machine learning models.

Concept drift can happen suddenly or gradually, and follow a periodic pattern where old concepts periodically reappear (e.g., circadian rhythm variations in neural firing rates). In such scenarios algorithms should remember previous contexts and re-instate them as soon as they reappear, overcoming catastrophic forgetting.

A basic strategy to address concept drift is to test for data distribution changes in data windows and retrain or update models when changes are detected. Several options exist for testing for distributional changes and for performing model updates. Alternatively, one could use ensemble methods that combine multiple models to mitigate the negative effects of drift (e.g., combine classifiers with different learning rates and weight them according to their accuracy. Most concept drift methods are designed for supervised learning, but methods such as clustering evolution, density estimation changes, and autoencoder-based monitoring can detect drift without labeled data.

Processing times for very large datasets

Neural and behavioral data analysis is most effective when computations are performed quickly, ideally in real time. Slow computations discourage data exploration and hinder scientific discovery. The large dataset sizes generated by NaLoDuCo experimentation pose a significant challenge for fast data analysis.

To overcome this limitation, we will combine distributed and GPU computing. Distributed computing is a paradigm in which tasks and data are divided across multiple computers. Instead of relying on a single powerful machine, distributed computing accelerates processing by executing multiple parts of a computation in parallel. GPU computing is a parallel computing approach that uses Graphics Processing Units (GPUs) to accelerate computational tasks. Unlike traditional Central Processing Units (CPUs), which execute a few complex operations sequentially, GPUs consist of thousands of smaller cores optimized for executing many operations simultaneously.

Distributed and GPU computing address different bottlenecks in large-scale computation. GPUs are highly efficient at parallelizing operations within a single machine. They excel at matrix operations and batch processing. However, GPUs are limited by memory and cannot scale indefinitely when dealing with huge datasets that exceed the GPU memory. Distributed computing allows to split workload across multiple machines, overcoming memory and computational limitations. It is particularly useful for scaling to massive datasets (e.g., long-term time series recordings).

We will develop accelerated implementations of all methods in the library of methods to process NaLoDuCo experimental data (Section 2.1). These implementations will use JAX[1] for model learning, inference, and numerical computation, Apache Spark[2] or Dask[3] to distribute pre-processing and feature extraction, and Ray[4] to distribute machine learning and deep learning functionality.

[1] https://docs.jax.dev/
[2] https://spark.apache.org/
[3] https://www.dask.org/
[4] https://docs.ray.io/

Thunder is a library develope in 2014 to accelerate the analysis of large scale neural data. It was introducing the use of distributed computing in neural data analysis. Our library is different from Thunder in that, besides analyzing large scale neural data, it processes continual recordings, and needs to overcome non-stationarity problems. In addition, it includes several methods to characterize behavior, while Thunder only targets neural activity. Finally, Thunder implements simpler methods assuming independent and identically distributed data, while our library contains more sophisticated time series methods.

2.1.3 Opportunities

More expressive models Our long-duration recordings, spanning weeks to months and generating hundreds of terabytes per experiment, will be transformative for neuroscience, much like the advent of large-scale datasets in computer vision. Just as the creation of MNIST—and later, ImageNet—enabled the training of deeper neural networks, leading to unprecedented performance breakthroughs, our massive, high-resolution neural and behavioural datasets will allow the estimation of far more expressive models than previously possible. For instance, large NaLoDuCo datasets will allow to estimate latent variable models with highly nonlinear and expressive models of observation given latent variables using recognition parametrised models (**?**). By capturing neural dynamics over extended timescales, we may uncover novel insights into learning, memory, and long-term neural adaptations that remain inaccessible with conventional short-duration studies.

Study very slow behavioural and neural rhythmic patterns Continuously monitoring detailed behavioural and neural activity over weeks to months enables the study of slow rhythmic processes that extend beyond traditional circadian (24-hour) rhythms, including ultradian (hours), infradian (days to weeks), and even multi-month cycles. These long-duration fluctuations influence learning, memory consolidation, motivation, and cognitive function, yet they remain largely unexplored in controlled experiments. By capturing these dynamics, we can gain new insights into neural plasticity, attention, and mood regulation, as well as the progression of neurological disorders like Parkinson's disease and depression, which exhibit slow symptom fluctuations.

New neuromodulation opportunities In traditional short-duration experiments, the effects of neuromodulation are tested immediately. In contrast, our long-duration experiments will enable repeated neuromodulation over extended periods and allow us to assess its impact over much longer timescales. For instance, in a mouse model of Alzheimer's disease, we could apply optogenetic stimulation to the hippocampus for one hour per day over the course of a month and assess its impact on memory retention and synaptic plasticity in the following weeks. This approach could reveal whether intermittent neuromodulation promotes long-term neural circuit stability and delays cognitive decline.

2.1.4 Related research

Neural data analysis methods from the Gatsby Unit The Gatsby Unit has developed world-class neural data analysis methods for inferring latent variables using Gaussian processes (**????**), or variants of linear dynamical systems (**??**), or recognition parameterised models (**?**), for separating contributions of different factors to spiking activity using tensor decompositions (**?**), and for understanding the effects of neural perturbations (**?**), just to mention a few.

Distributed computing for small animal Neuroscience **?** introduced a package[5] for analysing two-photon imaging records on distributed computing platforms. This has been a pioneering development by introducing distributing computing into neuroscience data analysis. However, it used short duration imaging recordings and it implemented simple data analysis methods.

Continuous epilepsy monitoring Continuous epilepsy monitoring is a transformative technology for diagnosing, understanding, and managing epilepsy. By capturing long-term brain activity, it provides crucial insights into seizure patterns and underlying neural dynamics.

A major advancement in this field is the development of implantable devices, such as the NeuroPace Responsive Neurostimulation (RNS) system, which continuously records electrocorticographic (ECoG) brain activity over

[5]https://github.com/thunder-project/thunder

extended periods (e.g., years). In addition to monitoring, RNS delivers targeted electrical stimulation in response to detected seizure precursors, significantly improving epilepsy treatment outcomes (**?**). However, RNS has two key limitations: (1) it is invasive, requiring surgical implantation, and (2) it can only store a limited amount of brain activity (typically a few hours) for later analysis, restricting its utility for studying long-term neural dynamics.

To overcome these challenges, researchers have developed subscalp ultralong EEG recording technologies (**?**), which use electrodes implanted under the scalp but above the skull. This approach is less invasive than intracranial devices and offers continuous EEG monitoring, with the added benefit of streaming data directly to the cloud. This capability enables long-term characterization of brain activity.

Subscalp EEG monitoring has already provided valuable insights into epilepsy. For example, it has revealed that seizure susceptibility is often modulated by circadian and ultradian rhythms, with specific times of day associated with increased seizure likelihood. However, compared to the AEON platform, subscalp EEG provides lower-resolution measurements of both neural and behavioural activity. Additionally, the data modeling methods used in subscalp EEG research remain largely proprietary, as much of this technology is developed by private companies. Furthermore, current implementations are primarily designed for seizure detection and forecasting, rather than for broader investigations into long-term brain dynamics.

2.1.5 Outputs

1. repository containing implementations of machine learning algorithms for offline processing NaLoDuCo experimental data, adapted to operate in non-stationary environments, and optimized to perform at scale when running on public clouds or institutional high-performance-computing clusters.

2. SWC NaLoDuCo foraging dataset storred in DANDI.

3. deployment of the methods in 1 in Amazon EC2 instances, to allow users to analyze on the cloud the datasets in 2.

2.2 Visual Exploration

2.2.1 Outputs

1. visualisations for continuous behavioural and neural recording
2. visualisations for epoched behavioural and neural recording
3. visualisations for model outputs
4. indexing system to support intelligent visualisations
5. deployment of the above items to allow users to visualise NaLoDuCo DANDI datasets on the cloud

2.3 Spike Sorting

2.3.1 Outputs

1. Repository with implementations and benchmarking of offline spike sorting algorithms for long-duration recordings
2. Repository with implementations and benchmarking of online spike sorting algorithms

2.4 Online Machine Learning

2.4.1 Outputs

1. Bonsai packages implementing real-time ML functionality for experimetal control
2. Documentation of these packages