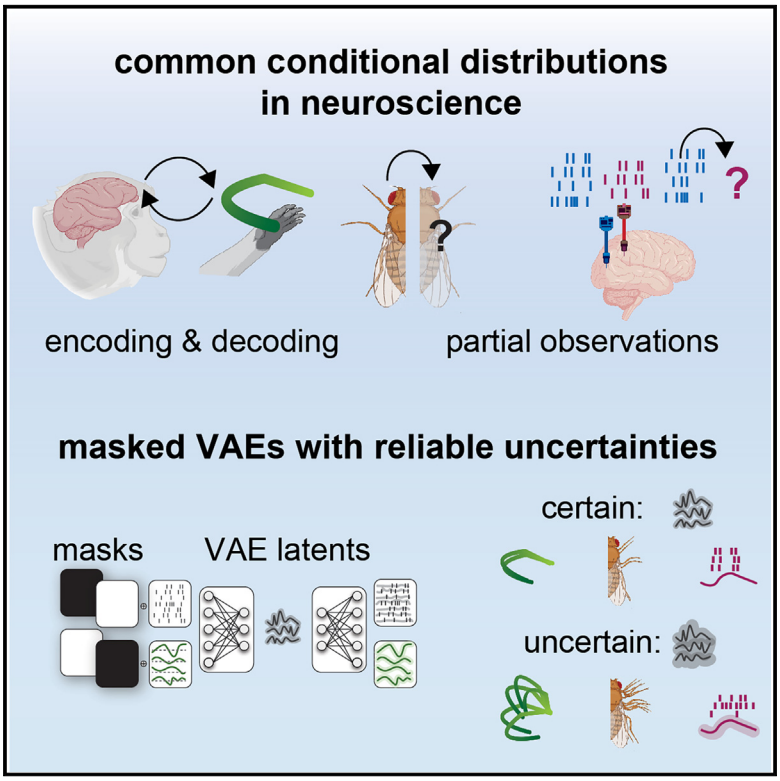
Cell Reports

Resource

# Modeling conditional distributions of neural and behavioral data with masked variational autoencoders

## Graphical abstract




## Authors

Auguste Schulz, Julius Vetter, Richard Gao, ..., Pavan Ramdya, Pedro J. Gonçalves, Jakob H. Macke

## Correspondence

auguste.schulz@uni-tuebingen.de (A.S.), jakob.macke@uni-tuebingen.de (J.H.M.)

## In brief

Schulz et al. demonstrate how neural encoding and decoding can be cast as computing conditional distributions and how to modify variational autoencoders (VAEs) to calculate such distributions. The proposed VAE-masking scheme allows for joint dimensionality reduction of neural and behavioral data and conditional generation of one modality given the other.

## Highlights

- We can recast neural encoding and decoding as computing conditional distributions
- We modify VAEs so that they can flexibly compute such conditional distributions
- Our method can be applied to various datasets, tasks, conditions, and levels of missingness
- Our approach results in more reliable uncertainty estimates than standard VAEs

Schulz et al., 2025, Cell Reports *44*, 115338
March 25, 2025 
https://doi.org/10.1016/j.celrep.2025.115338

CellPress

Cell Reports

CellPress
OPEN ACCESS

Resource

# Modeling conditional distributions of neural and behavioral data with masked variational autoencoders

Auguste Schulz,[1,9,*] Julius Vetter,[1] Richard Gao,[1] Daniel Morales,[2] Victor Lobato-Rios,[2] Pavan Ramdya,[2] Pedro J. Gonçalves,[1,3,4,5,6,8] and Jakob H. Macke[1,7,8,*]

[1]Machine Learning in Science, University of Tübingen & Tübingen AI Center, Tübingen, Germany
[2]Neuroengineering Laboratory, Brain Mind Institute & Interfaculty Institute of Bioengineering, EPFL, Lausanne, Switzerland
[3]VIB-Neuroelectronics Research Flanders (NERF), Leuven, Belgium
[4]Imec, Leuven, Belgium
[5]Department of Computer Science, KU Leuven, Leuven, Belgium
[6]Department of Electrical Engineering, KU Leuven, Leuven, Belgium
[7]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[8]These authors contributed equally
[9]Lead contact
*Correspondence: auguste.schulz@uni-tuebingen.de (A.S.), jakob.macke@uni-tuebingen.de (J.H.M.)
https://doi.org/10.1016/j.celrep.2025.115338

## SUMMARY

Extracting the relationship between high-dimensional neural recordings and complex behavior is a ubiquitous problem in neuroscience. Encoding and decoding models target the conditional distribution of neural activity given behavior and vice versa, while dimensionality reduction techniques extract low-dimensional representations thereof. Variational autoencoders (VAEs) are flexible tools for inferring such low-dimensional embeddings but struggle to accurately model arbitrary conditional distributions such as those arising in neural encoding and decoding, let alone simultaneously. Here, we present a VAE-based approach for calculating such conditional distributions. We first validate our approach on a task with known ground truth. Next, we retrieve conditional distributions over masked body parts of walking flies. Finally, we decode motor trajectories from neural activity in a monkey-reach task and query the same VAE for the encoding distribution. Our approach unifies dimensionality reduction and learning conditional distributions, allowing the scaling of common analyses in neuroscience to today's high-dimensional multi-modal datasets.

## INTRODUCTION

Recent developments in experimental techniques allow real-time behavioral tracking of animals[1–3] and simultaneous recordings of hundreds of neurons across multiple brain regions.[4–6] Modern datasets in neuroscience are thus increasingly large, high-dimensional,[7,8] and commonly consist of multiple modalities—e.g., neural activity and behavior[9]—that often have highly non-linear relationships.[10] While data collection has changed drastically in the last years, an important goal of systems neuroscience remains the same: understanding how brain activity gives rise to complex behavior.

To gain insights from neural and behavioral data, neuroscientists have developed various neural encoding and decoding models.[11] These tasks should ideally be addressed in a probabilistic manner to account for the inherent variability of neural and behavioral measurements and in order to quantify resulting uncertainty. As experimental neuroscience is moving toward less controlled, unconstrained, multi-modal data collection, this aspect becomes even more relevant. Both probabilistic encoding and decoding tasks can, algorithmically, be boiled down to the task of calculating conditional distributions: encoding studies in neuroscience involve calculating the conditional distribution of neural activity given behavior or other observations such as stimuli.[12] Conversely, for decoding analyses, one needs to calculate the conditional distribution over behavior, given neural activity (Figure 1A, left). Generating interpretable, accessible neuroscientific predictions from complex, high-dimensional data directly is very challenging,[13,14] highlighting the need for tools that can infer low-dimensional representations of high-dimensional neural and behavioral datasets (Figure 1A, right). In short, to gain neuroscientific insights from such complex datasets, our goal is to unify (1) the ability to link neural and behavioral data (i.e., through encoding/decoding models; Figure 1A, left), and (2) joint dimensionality reduction of the data, ideally in a probabilistic and generative manner (Figure 1A, right).

Various dimensionality reduction methods have demonstrated that a substantial fraction of variability both in unconstrained behavior and neural population activity can be captured by a few latent (i.e., unobserved) dimensions.[15–21] This insight has driven the development of various latent variable models

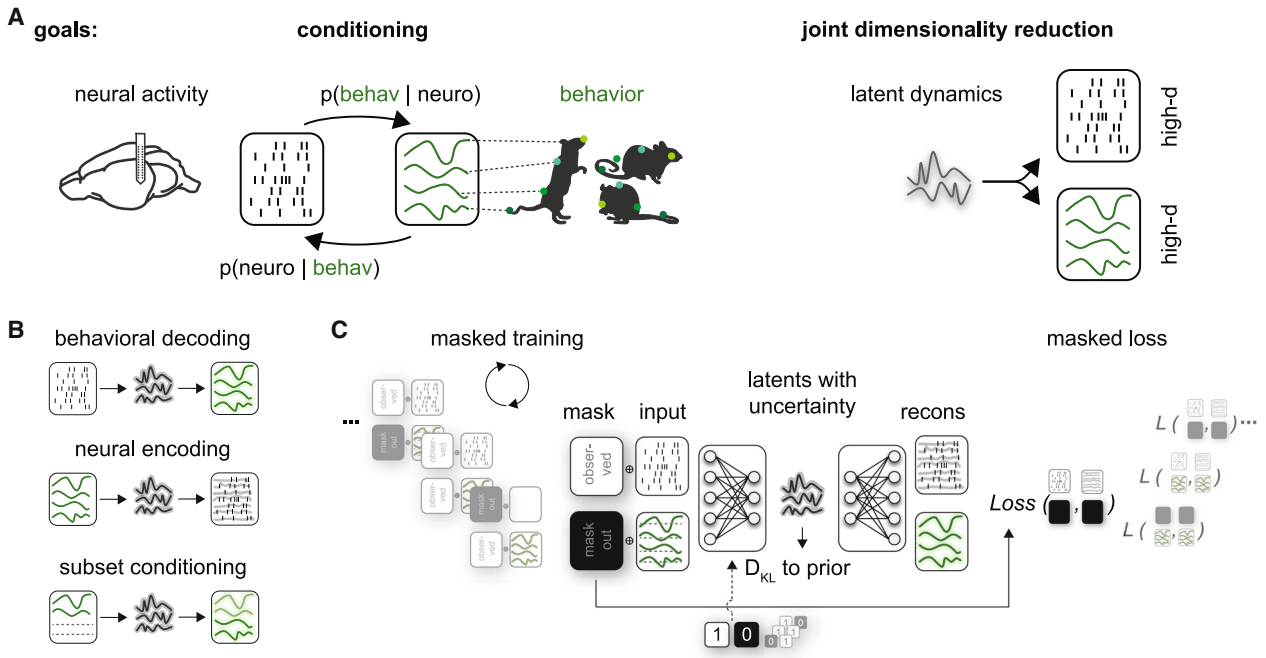



**Figure 1. Latent variable models that can deal with conditional distributions arising in large-scale multi-modal datasets**

(A) Our approach aims to address two common goals of (neuro)scientific analyses: conditioning, e.g., for linking brain activity (neuro) and behavior (behav) and joint dimensionality reduction of potentially high-dimensional neural and behavioral data.

(B) Conditional distributions arise, e.g., when learning the distribution over behavior given brain activity (behavioral decoding), the distribution over neural activity given behavior (neural encoding), or when analyzing the interaction of data subsets such as different behavioral variables or multiple brain regions.

(C) Masked variational autoencoder training scheme for data of potentially different data types (e.g., count and continuous) with structured masks for modeling conditional distributions. During each training iteration (for which a subset of the training data is passed to the network), one mask is chosen randomly, and the corresponding inputs to the network are replaced by a fixed imputation value (zero or the mean). The reconstruction loss is computed solely on observed data, i.e., for different masks, the reconstruction loss is computed on a distinct subset of the data. Identical to classical VAEs, the training objective combines the (masked) reconstruction loss and a regularization term for the latent space: the Kullback-Leibler divergence $D_{KL}$ of the inferred latent representation and a defined prior (here, standard Gaussian). Optionally, one can pass the binary mask (1, observed; 0, masked) to the encoder network (dashed line, see STAR Methods).

that infer underlying low-dimensional representations from neural data.[16,22–26] Classical methods often require simplifying modeling assumptions, such as (switching) linear dynamics,[27–29] or strictly Gaussian observations,[15] and rely on model-specific optimization schemes (e.g., expectation-maximization algorithms or subspace-identification methods).[19,30] With the rise of deep learning and more flexible optimization schemes, various of these assumptions can be relaxed,[16,20,21] leading to latent variable models that can capture complicated non-linear relationships in the data and underlying low-dimensional dynamics.[31]

One such model class, exploiting deep inference networks, is the variational autoencoder (VAE).[32,33] Inference networks of VAEs take observed data as the input and return a distribution over the latent state. VAEs are, however, often primarily used as tools for dimensionality reduction, where data are compressed and then decompressed. However, VAEs provide a full, probabilistic, non-linear latent variable model that can approximate the whole underlying latent distributions rather than providing only a compressed point estimate. Sequential variants of VAEs can infer latent representations underlying heterogeneous time-series datasets, e.g., consisting of both continuous and count data (spiking),[31,34,35] and are commonly used in analyzing neural and behavioral data.[16,20,23,25,36] However, most VAE-based methods cannot adequately deal with a ubiquitous analysis task in neuroscience: calculating arbitrary conditional distributions $p$(data subset A|data subset B).[37–39] The reason for this is that inference networks of VAEs typically can only deal with *fully* observed input data, and have no means to model the (additional) uncertainty arising from partial observations. Such conditional distributions do not only arise when estimating behavioral decoding (Figure 1B, top) and neural encoding distributions (Figure 1B, middle) but also become relevant when dealing with partially observed data or when studying interactions between brain regions or tracked body parts (Figure 1B, bottom). An ideal model should correctly estimate how uncertain it is about the inferred underlying representation and its predictions (error bands, Figure 1B). Accurate uncertainty estimates can tell us how constrained one subset is given the other subset and let us reason about their dependencies beyond accuracy or similarity scores. However, in neuroscience, the quality of uncertainty estimates is typically not assessed.

In this work, we present an approach that enables VAEs to accurately model arbitrary data conditionals arising in neuroscience. Specifically, we use a masked-training approach and demonstrate it in a variety of neuroscience applications where we achieve both dimensionality reduction and sampling

of conditional distributions. Furthermore, we propose calibration tests to assess the quality of the generated conditional distributions.

VAEs have been extended to model the distribution of a missing data subset given an observed data subset.[34,39] Conceptually, we build on these approaches and treat the modality we want to learn the conditional distribution over as missing. For example, to capture a behavioral decoding distribution, we modify the loss and training scheme of a VAE during joint training on neural and behavioral data by stochastically masking behavior. Our training approach is not limited to a specific modeling architecture but can be applied to a variety of VAE approaches. We showcase our approach on diverse datasets: sequential and static, multi- and uni-modality, discrete and continuous datasets, each of which has a different VAE architecture. We first validate our approach on a task on which we have access to the ground-truth distributions. We demonstrate that our approach allows for correct inference of low-dimensional latents and accurate predictions. On a high-dimensional behavioral dataset of walking flies, we successfully recover the relationship between different body parts along with uncertainties and obtain realistic samples from the conditional distributions of masked legs. Finally, we showcase the approach in a challenging multi-modal neural and behavioral dataset, where we model encoding and decoding distributions of high-dimensional population activity from primary motor areas and self-paced reach movements.[40] Neural latents extracted from partial neural recordings reveal that our masked VAE approach has the desired property of increasing uncertainty when predictions are likely wrong.

## RESULTS

### Masked training of variational autoencoders for estimating conditional distributions

To prepare variational autoencoders to deal with conditional distributions commonly arising in neuroscience, we modify the training scheme of classical VAEs. To reiterate, VAEs can approximate the whole underlying latent distribution, often parametrized by a mean and variance, rather than providing point estimates. This aspect is critical for handling partial or masked observations and corresponding uncertainty levels in the latent distribution. During joint training on multiple data subsets, our approach prepares the network for each subset to be structurally masked at test time (Figure 1C). We use the term structured masking to refer to algorithmic masking of data subsets for conditioning to avoid confusion with actual data missingness—e.g., individual input channels that drop in and out. First, we specify the structured masks depending on the desired conditional distributions and specify how often (on average) each mask should be selected during training (Figure 1C, left; see STAR Methods). During each training iteration, a random subset of the training data, often referred to as a mini-batch, is passed to the network, and one mask is chosen randomly. The corresponding masked inputs are replaced by a fixed imputation value (e.g., zero or the mean). We then calculate the reconstruction loss $\mathcal{L}$ solely considering observed subsets (see STAR Methods) in the evidence lower bound (ELBO), which is the optimization target of VAEs.[33,39] The ELBO combines such reconstruction terms with a regularization term in latent space. Here, we compute the Kullback-Leibler divergence $D_{KL}$ between the inferred latent representation and a standard Gaussian prior over the latents. The training objective of masked VAEs, thus, only differs from classical VAEs in terms of the masked reconstruction loss. Passing the mask to the reconstruction loss is the crucial component that instructs masked VAEs to update their weights to appropriately deal with different conditional distributions. Additionally, to make it easier for the network to learn that an input has been masked, one can pass the binary mask to the encoder network, in addition to the masked information in the loss. In cases where dimensions masked during training would, at test time, happen to take the same value as the imputation value (e.g., silent neural populations and zero imputation), the binary mask should always be passed. This allows the VAE to properly disentangle whether these dimensions are masked or observed.

In summary, we propose modeling conditional distributions with VAEs, e.g., for neural encoding and behavioral decoding, by recasting it as a structured masking problem. This approach allows us to sample from a distribution of interest, e.g., to visualize time series of various potential behaviors that are likely given a neural population activity trace and vice versa. As such, our results demonstrate, in various application scenarios, how to perform latent inference and generate samples of data modalities that are unobserved at test time. The generality of this approach allows for applying it to a variety of conditional distributions and variational autoencoder settings.

### Inference of conditionals in a tractable Gaussian latent variable model

First, we evaluated whether our training scheme and loss modification allow us to learn the correct distributions of interest on a simulated dataset where we have access to the ground-truth conditional and posterior distributions. This dataset was generated from a Gaussian latent variable model (GLVM) with latent (unobserved) random variable $z$ and data dimensions $x$, which linearly depend on the latent $z$ (Figure 2A; see STAR Methods). In this illustrative example, we can think of a subset of $x$ as the high-dimensional neural activity, another subset of $x$ as high-dimensional behavior, and the latent variable $z$ as the low-dimensional representation underlying both neural and behavioral data. The inference network infers the distribution over these unobserved latents given a chosen $x$—i.e., it calculates the posterior distribution $p(z|\text{observed x})$—effectively inverting the data-generation process in a probabilistic way. The strength of the linear coupling and the noise levels of individual $x$ dimensions define how much information about the latent can be gained by observing those $x$ dimensions.

We contrast the masked VAE with a regular VAE trained on all data (referred to as naive training) regarding the capacity to capture data conditionals $p(\text{masked x}|\text{observed x})$ at test time (Figure 2). The naive approach fails to capture the true data distribution (gray), with overly narrow 1D and 2D (marginal) distributions (Figure 2B, left). Masked VAEs, however, can successfully reconstruct observed values ($x_i^{obs}$) and impute masked ones ($x_i^{unobs}$) (Figure 2B, right). The reason for this discrepancy lies in the ability to learn the distribution over the latents (posterior distribution) when some of the input data are unobserved.

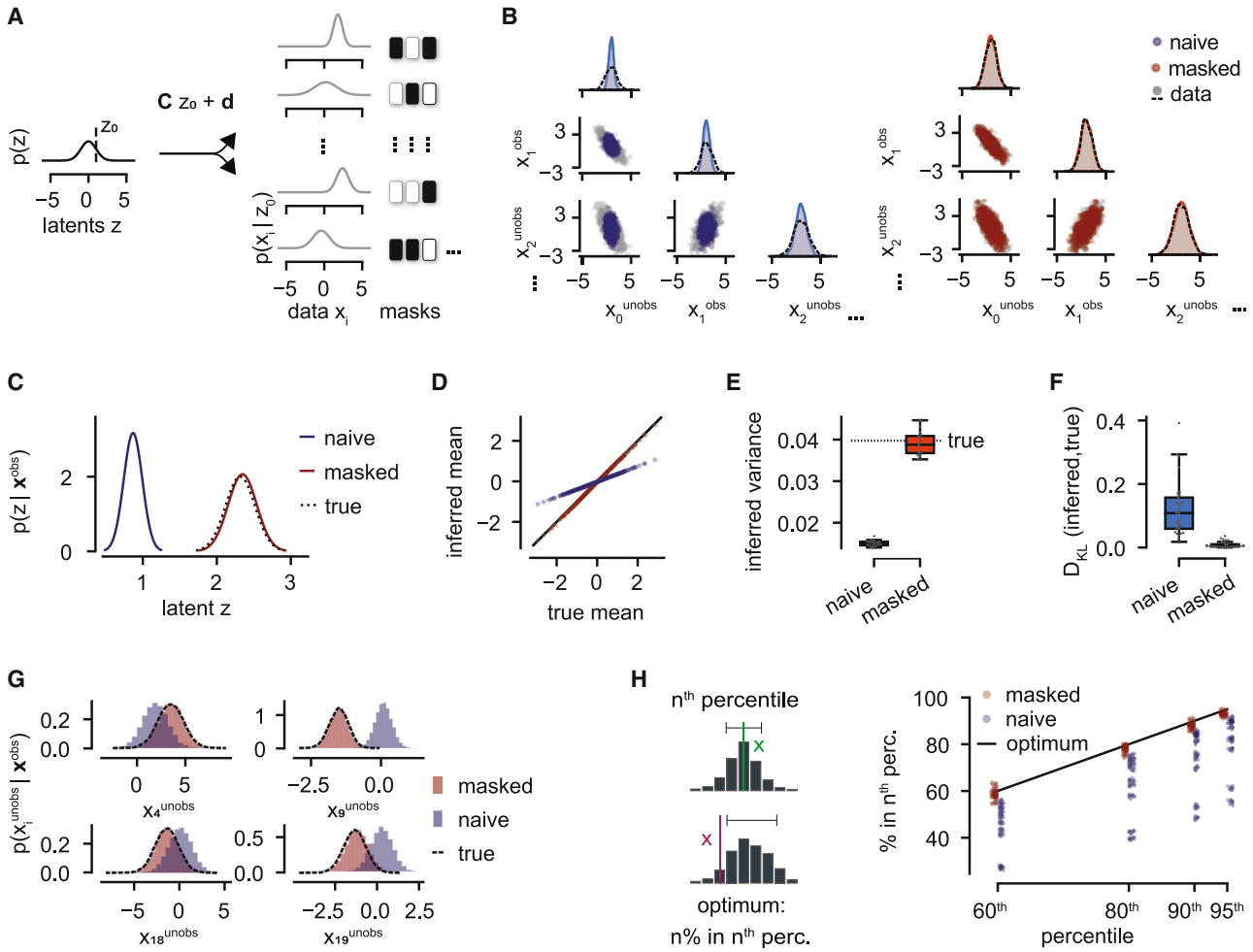



**Figure 2. Inference of conditionals in a GLVM**

(A) Left: schematic of a GLVM with fixed parameters $\theta = \{C,d,\Lambda\}$, where $\Lambda = \mathrm{diag}(\sigma_i^2)$ for $i \in \{1,20\}$. Right: masks for conditioning specified by the user, where each column is a different mask. Here, 50% of the values are masked.

(B–H) Comparison of our masked-training approach (red) with a vanilla VAE approach (blue, naive) when half of the inputs $x$ are masked out at test time.

(B) Model reconstructions and true test data samples (1D and 2D marginal distributions over $x$).

(C) Inferred and analytical (true) posterior distributions over latent z given only the observed dimension of one test sample, i.e., $p(z|\mathbf{x}^{\mathrm{obs}})$.

(D) True versus inferred posterior mean given a range of test samples.

(E) Inferred posterior variance across multiple instantiations (seeds) and true posterior variance (dotted line). Data are represented as mean over test samples. Boxplots show the median and lower and upper quartiles.

(F) Average Kullback-Leibler divergence between true and inferred posterior distributions across different GLVM parameters ($\theta = \{C,d,\Lambda\}$) and structured masks. Data representation as in (E).

(G) Conditional distributions over randomly chosen masked $x$ dimensions (see STAR Methods) for the same test sample as in (C).

(H) Left: schematic of statistical calibration, evaluating the quality of uncertainty estimates. Optimal calibration: n% of true data points lie in the $n^{\mathrm{th}}$ percentile confidence interval of the sampling distribution. Example of an $x$ within the interval (green, top) and one outside of it (red, bottom). Right: calibration checks of predicted conditional distributions $p(x_i^{\mathrm{unobs}}|\mathbf{x}^{\mathrm{obs}})$ for all masked $x$ dimensions across multiple model seeds. See also Figures S1-S4.

Masked training can perfectly infer the true analytically calculated posterior. In contrast, naive training fails to do so (Figure 2C). The naive network does not detect masked values as such. Hence, its posterior mean inference is biased (Figure 2D), and the posterior variance is too small (Figure 2E). These observed discrepancies between masked and naive VAEs are even more pronounced when more than 50% of the values are masked. Conversely, when only one of the dimensions was masked, the discrepancies were less pronounced (Figure S1). In short, naive VAE training leads to confidently wrong predictions, while the masked network correctly adjusts the uncertainty about its predictions. This finding generalizes across different parameter sets ($C, d, \Lambda$) and masking conditions, and observation noise ranges (Figures 2F, S2, S3, and S4). The higher the overall observation noise, however, the more training samples are required to achieve a good model fit (Figure S3).

One of the advantages of VAEs is that, once trained, sampling from VAEs is straightforward. Thus, we can easily investigate

CellPress
OPEN ACCESS

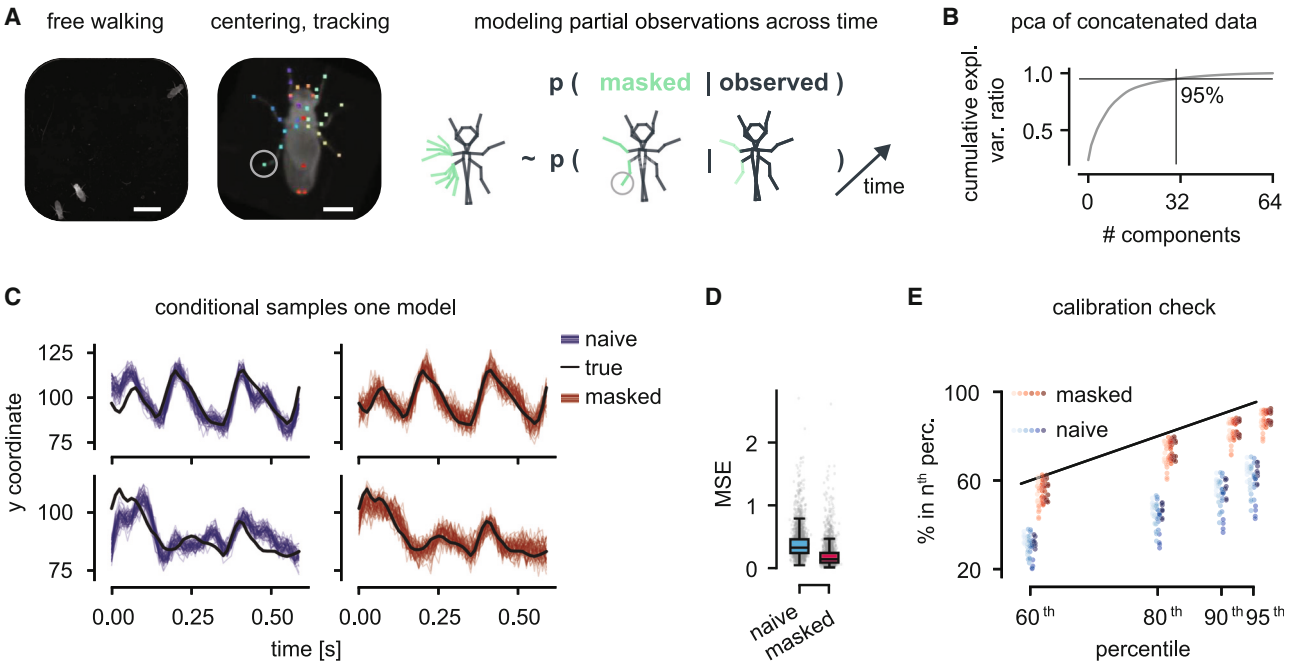



**Figure 3. Conditional sampling of masked legs of walking flies**

(A) Three flies are filmed from the bottom during walking behavior in a constrained arena. The scale bar represents 5 mm. Cropped video frames of individual flies are centered, aligned for constant head direction, and tracked with DeepLabCut, resulting in a 64-dimensional time series. The scale bar represents 1 mm. Schematic of the target distribution: the conditional distribution over masked left legs given the remaining body key points.

(B) Cumulative explained variance of the number of components when performing principal-component analysis on all concatenated time-series data.

(C) Two example time series (from the test set) of the unobserved limb, marked with a circle in (A). Conditional samples from the masked model in red, naive in blue, and true limb trajectory in black.

(D) Mean squared error (MSE) of mean predictions for the masked limb, averaged across time and test samples, for both training schemes. Boxplots show the median and lower and upper quartiles.

(E) Calibration checks of predicted conditional distributions $p(\text{limb}_i^{\text{unobs}} | \text{limbs}^{\text{obs}})$ for all masked leg key points shown in (A), across multiple model seeds (different hue per seed). Optimal calibration in black.

whether the conditional samples for unseen test data correspond to the conditional distribution we set out to model. Wrongly inferred posterior distributions will likely result in inaccurate conditional samples.

Indeed, for four example masked data dimensions ($x_i^{unobs}$), the distribution of conditional samples from the naive VAE is wrong, whereas the masked training distribution matches the true (analytical) conditional (Figure 2G).

In real neuroscientific datasets, we do not have access to the true conditional distributions for comparison. For such cases, we propose to evaluate the quality of the inference method and its uncertainty estimates with an adaptation of simulation-based calibration.[41,42] These calibration checks allow us to also evaluate the quality of the uncertainty estimates and thus go beyond the evaluation of mean squared error or log-likelihood of structurally masked values, which informs only about the quality of the mean predictions. More concretely, calibration checks count how often masked test sample values $x_i^{unobs}$ lie in the respective predicted conditional sampling distribution (Figure 2H; see STAR Methods). We found that, for the masked training, predictions are well calibrated, i.e., neither overconfident nor underconfident (values close to the diagonal Figure 2H, red). In contrast, the naive approach is confidently wrong for many test cases (Figure 2H, blue). Overall, these results suggest that masked training allows us to infer both the correct posterior $p(z|\text{only observed x})$ and conditional distributions $p(\text{masked x}|\text{only observed x})$ in a tractable, well-specified example. The results demonstrate the statistical challenges that arise when one aims to use VAEs to perform conditioning and provide a theoretical basis for applying the masked-training approach to neuroscientific data.

## Probabilistic conditional modeling of masked key point trajectories in fly walking behavior

Next, we wanted to investigate whether the masked approach is applicable to complex time-series data in neuroscience and can successfully model conditional distributions of scientific interest. In particular, we focused on an experiment that characterizes the (backward) walking behavior of the fruit fly, *Drosophila melanogaster*, and applied our masked-training approach to a sequential VAE developed for this high-dimensional behavioral dataset.

We obtained the dataset by tracking the centroids of individual flies and aligning the video frames such that fly heads are all pointing upward (Figure 3A). We tracked 32 body parts ($x$, $y$ each) with DeepLabCut,[1] resulting in a 64-dimensional time series (see STAR Methods). To account for the temporal structure

of the data, the VAE's architecture has both convolutional elements and recurrent neural networks based on gated recurrent units,[43] as well as elements for non-linear dimensionality reduction (see STAR Methods). Analogous to the GLVM case, we adapted the masked-training scheme for this sequential VAE to allow for modeling the conditional distribution over a subset of the fly body key points, given the remaining ones (Figure 3A, right, masked legs in green; see STAR Methods). Here, we chose to mask body key points that are crucial for walking behaviors and show characteristic variability during walking: hind claw, hind tibia-tarsal joint, mid tibia tarsus, and mid claw of the left side.

If the model captures the dependence correctly via the compact latent representation, we expect accurate conditional modeling of these masked legs. In the previous example, the variability in the data was captured by one latent variable, but, for experimental data, the underlying dimensionality is unknown. Here, we are dealing with a dataset with high intrinsic dimensionality: almost 32 principal components are required to capture 95% of the variance of the 64-dimensional dataset (Figure 3B). In our sequential VAE, we can exploit temporal dependencies and thus further reduce the dimensionality of the latent space while capturing stereotyped walking behavior (see STAR Methods). Samples from the naive model capture overall trends of the masked left hind claw (circle in Figure 3A) well, particularly during highly periodic walking (Figure 3C, left). However, ground-truth trajectories often deviate from the sampled trajectories (blue). In contrast, masked training produces more faithful predictions and uncertainty estimates (red), reflected in the inclusion of the ground truth for most time steps (Figure 3C, right). This is also captured by a lower average mean-squared error (MSE) of the masked VAE test predictions (averaged across time) of the masked key point shown in Figure 3C. However, MSE alone does not immediately reveal a substantial performance boost through masked training (Figure 3D). The difference, however, becomes clear when inspecting the uncertainty estimates: when the naive approach is wrong, it is confidently wrong (large deviations from the diagonal in Figure 3E), while the masked approach is better calibrated.

We conclude that our masked-training methodology is indeed applicable to time-series datasets and allows us to faithfully model the conditional distributions of masked body key points given the remaining ones. Masked training leads to better uncertainty estimates—it allows the network to know better when it does not know.

## Decoding continuous reaches from neural population activity

To be effective for neuroscientific research, our method should be able to deal with data types and modalities that commonly arise in neuroscience. Therefore, we implemented the masked-training scheme for a classic monkey-reach task, which is particularly challenging due to its continuous instead of trial-based structure (using publicly available data from O'Doherty et al.,[40]; Figure 4A, left). We focus on the behavioral decoding distributions, i.e., the conditional distribution of $x$- and $y$-reach directions given activity traces of $>200$ neurons (Figure 4A, right).

The monkey reaches toward an indicated light target on an 8 × 8 grid, leading to movements of different lengths, heterogeneous angles, and velocities (Figure 4B). Neural activity is simultaneously recorded in primary motor cortex. The maximum spike count of individual units is six spikes in time windows of 64 ms (Figure 4C; binning consistent with Makin et al.[44] to capture behaviorally relevant timescales). We built a sequential VAE for this multi-modal dataset, which consists of both continuous (behavioral) and discrete data (spike counts). Our reconstruction loss, therefore, is composed of a Poisson- (for spike counts) and Gaussian- (for behavior) negative log-likelihood (GNLL) loss (see STAR Methods[34,35]). We specified masks for encoding and decoding distributions during training; i.e., we masked either neural activity (replaced by zeros) or behavior (replaced by the mean cursor position). Masked mean reconstructions (red) of behavioral traces are more accurate than naive (blue) predictions across many model seeds (Figure 4D). Surprisingly, the naive approach is performing relatively well, and, while we see some sections where the masked approach is performing better, the errors are quite consistent across the two approaches. This indicates that some sections of the traces are less correlated with neural activity than others, and both models are capable of exploiting some correlations required for conditional modeling. Sampling from individual masked and naive models (Figure 4E) and calibration checks (Figure 4F) again demonstrate that the masked but not naive approach targets the conditional—the decoding distribution. Note that a discrepancy between $x$ (bottom rows) and $y$ (top) decoding performance (Figures 4D–4F) has been reported previously for this dataset.[40,44,45] While the masked VAE is not perfectly calibrated on this dataset, it clearly outperforms the naive VAE.

In conclusion, our masked-training approach can be readily applied to multi-modality datasets and makes it possible to sample from a conditional distribution over a continuous time series given high-dimensional time series of discrete count data.

## Encoding of continuous reaches in neural population activity

Next, we assessed the performance of the same trained model on the reversed and more challenging task: modeling the high-dimensional conditional distribution over the activities of 213 neurons in primary motor cortex given only the two-dimensional behavioral trajectories (Figure 5A).

Both masked and naive approaches generally capture the histogram of observed spike counts given the reach trajectories (Figure 5B)—despite a slight over-prediction of spike counts—but the naive approach has worse population-average estimates across model instantiations (Figure 5C). The log-likelihood per neuron across model instantiations reveals the superior performance of the masked versus naive encoding (Figures 5D and 5E).

Samples from the trained models suggest that both masked and naive approaches correctly predict time-varying firing rates that clearly reflect the reach movements (Figure 5F). Notably, the masked approach reveals higher variability in the rate predictions reflecting higher posterior uncertainty.

Spike counts are discrete rather than continuous variables, so we adapted the method to assess the uncertainty calibration: we compare the cumulative distribution function (CDF) of the ground-truth spike train against a Poisson spike train with a

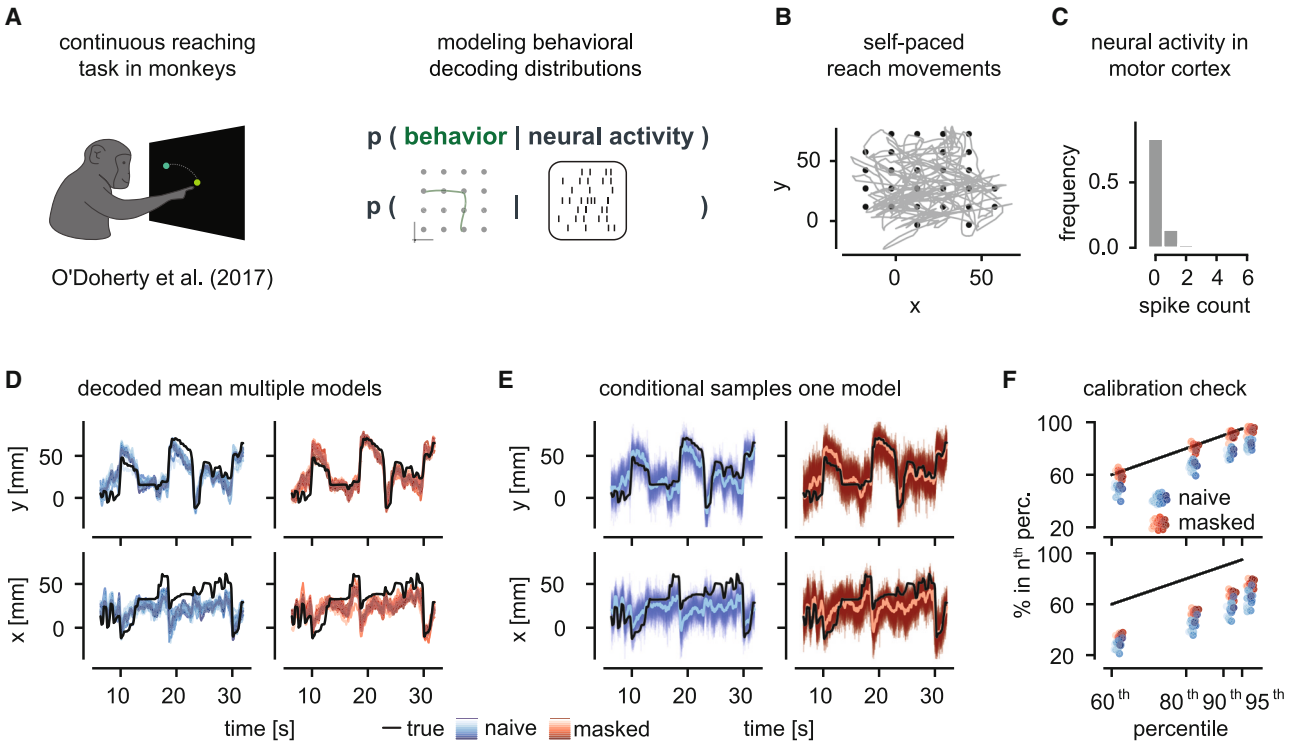



**Figure 4. Behavioral decoding of continuous reach movements from monkey primary motor cortex**

(A) A monkey performs self-paced, continuous reaches on an 8 × 8 grid with simultaneous cortical electrophysiology recordings. Schematic of the target distribution: the conditional distribution over behavior, in this case, cursor trajectories given neural population activity recorded in monkey primary motor cortex.

(B) Behavioral trace of continuous reach movements (gray) and targets (black).

(C) Frequency of observed spike counts across primary motor cortex during the reach movements shown in (B), binned at 64 ms.

(D) Example traces of cursor positions and the mean predictions for the naive (blue) and masked (red) modeling approaches for multiple model seeds (different hue per seed).

(E) Same cursor traces as in (D) but with conditional samples from a single naive (blue) and masked (red) model, respectively.

(F) Calibration checks of predicted conditional distributions $p(\text{behavior}|\text{neural activity})$ respectively for x-y directions, across multiple model seeds. Optimal calibration in black.

(D–F) The y direction is in the top and the x direction in the bottom row.

rate sampled from either the masked or the naive models (Figure 5G). We find that the sampling distributions for both masked and naive capture the ground-truth distribution reasonably well (Figure 5G; Figures S5-S8 for other units).

In conclusion, our masked VAE approach allows us to model and produce samples from both decoding and encoding distributions in one single model without requiring any retraining.

## Decoding from the latent space of partial neural recordings

Finally, we explored the relationship between latent uncertainty and downstream task performance, specifically decoding movements from latents extracted from partial neural observations (Figure 6A). We posit that, when performing downstream decoding from such latents, an important advantage of uncertainty estimates is that they can indicate when not to trust a decoded movement, namely when the corresponding uncertainty is too high.

In order to test this idea, we trained a VAE on neural activity alone. First, we specified different masking levels, ranging from five masked neurons out of 213 to 200 masked neurons. As before, we sampled the masks during masked training and passed masked spikes as zeros (Figure 6B). While masked VAEs increase the latent uncertainty of their most informative latents (highest mean variability across time; see STAR Methods) with increasing mask size, naive VAEs fail to do so (Figure 6C). This holds across different masking levels and throughout the test set (Figure 6D).

Second, we used linear ridge-regression models to decode movement velocity from either the latent means of the masked or naive VAEs, or directly from spikes (Figure 6E). We found that decoding performance (measured as the correlation between the true and predicted velocity) from raw spikes was lower than from inferred latents. This result is in agreement with previous literature,[44,46] further highlighting the advantage of VAE-based latent variable models.

Finally, we assessed how decoding performance is correlated with latent uncertainty. In order to accomplish that, we averaged the mean latent uncertainty over time for different masking levels for the masked and naive VAEs. We then fitted per VAE instantiation (seed) a linear regression from decoding performance to latent uncertainty across masks. The linear fits reveal a negative

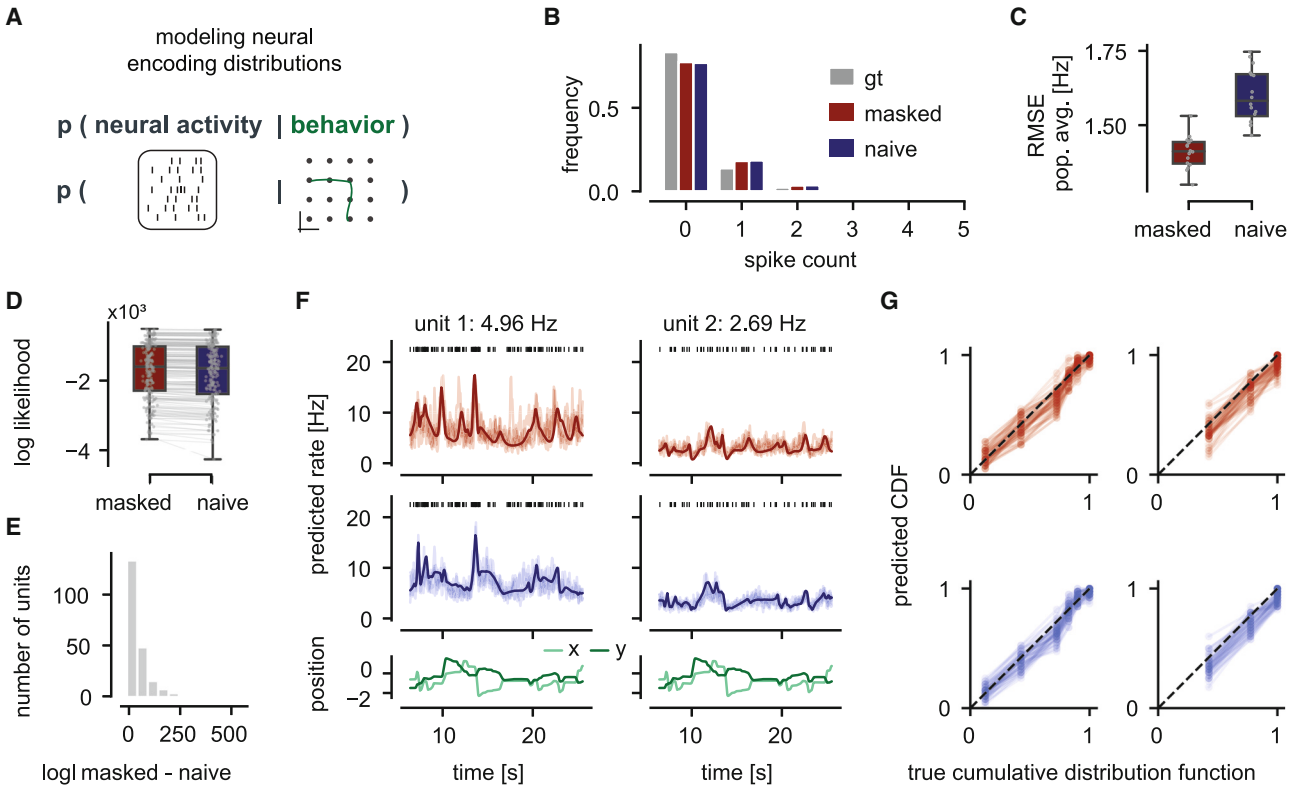



**Figure 5. Neural encoding distributions given continuous movements in the monkey-reach task**

(A) Schematic of the target distribution: the conditional distribution over neural population activity recorded in monkey primary motor cortex given behavior, i.e., cursor trajectories.

(B) Frequency of observed spike counts across primary motor cortex during reach movements binned at 64 ms and the predicted spike-count distribution of the masked (red) and naive (blue) approach.

(C) Root-mean-squared error (RMSE) of the population average and predicted population average for different model seeds. Data are represented as mean across time points. Boxplots show the median and lower and upper quartiles.

(D) Log-likelihood per neuron for the masked and naive approaches. Data are represented as mean across time points and model seeds. Higher is better. Boxplots as in (C).

(E) Distribution of differences in log-likelihood (logl) of the masked minus the naive approach. Positive values indicate a better model fit of the masked approach.

(F) Sampled rate predictions (10 each) and mean rate prediction from both models (masked red top, naive blue middle) for two example neurons with different activity levels given the standardized behavioral trajectory (bottom).

(G) CDF of the observed spikes (true) vs. predicted spike distributions sampled from the masked (red) and naive (blue) VAE for the rate predictions shown in (F). To calculate the CDF, spike counts are aggregated across five bins due to low spike counts. Optimal predictions would lie on the diagonal (black dotted line). See also Figures S5–S8.

correlation for masked, but not naive, VAEs, between uncertainty prediction and decoding performance (Figure 6E): for the masked case, 10 out of 10 model instantiations reveal a significant ($p < 0.005$) negative slope, with values ranging from −2.8 to −1.5 (R-squared between 0.84 and 0.98). For the naive case, we obtain slopes ranging from 0.04 to 0.22 (R-squared between 0.11 and 0.87), where many (six out of 10) are not significantly different from 0 ($p > 0.05$).

In conclusion, we find that masked but not naive VAEs have the desired property of increasing uncertainty when the predictions are likely wrong (low decoding performance).

## DISCUSSION

We introduce a training methodology for modeling conditional distributions with masked variational autoencoders, bridging dimensionality reduction, generative modeling, and encoding and decoding analyses in neuroscience. Our experiments show that modifying the training scheme and loss through structured masking enables VAEs to model specified conditional distributions. Thus, our approach allows for joint dimensionality reduction of high-dimensional multi-modal data and conditioning on specified modalities. It is not restricted to specific architectural or modeling choices and can be easily applied to a variety of variational autoencoders deployed in neuroscience. We validated our approach on a tractable example in which we correctly learned the ground-truth posterior and conditional distributions. We applied our approach to two neuroscientific time-series datasets: a continuous reach task in monkeys,[40] in which we probabilistically encoded behavior in—and decoded behavior from—high-dimensional neural activity, as well as a behavioral dataset of walking flies, for which we successfully modeled the

CellPress
OPEN ACCESS

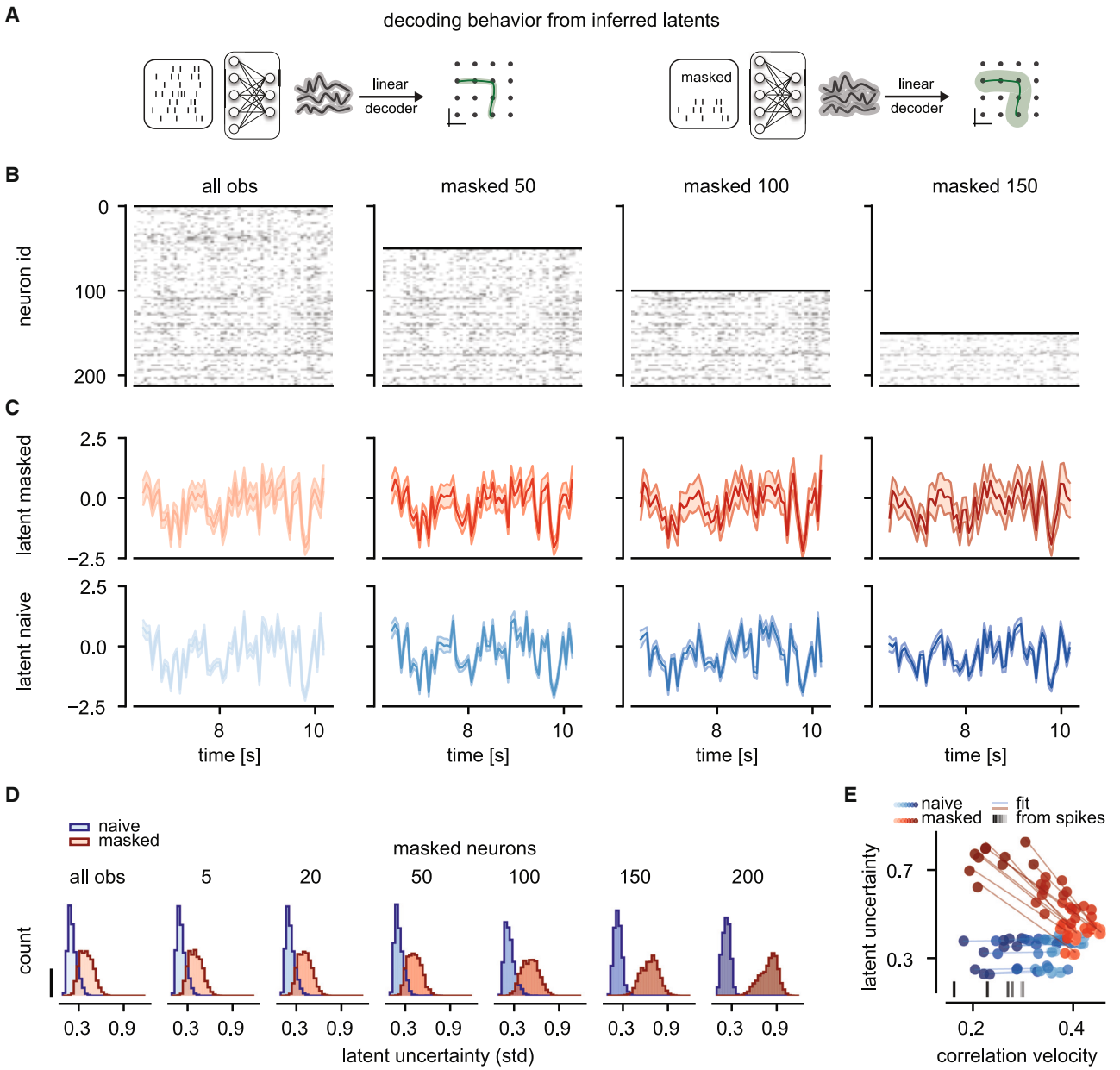



**Figure 6. Impact of partial neural observations on latent uncertainty and downstream decoding**

(A) Decoding from uncertainty-aware latents: latent uncertainty should increase for partial (right) vs. full (left) observations, which would translate to more variable decoded behavior.

(B) Spiking data for fully observed and masked conditions with three masking levels. Masked spikes are passed as zero values.

(C) Latent state over time represented as mean ± standard deviation (shaded region) returned by the VAEs for masked (red) and naive (blue) VAEs.

(D) Distribution of latent uncertainty per time step at varying masking levels (all observed, and masking of 5, 20, 50, 100, 150, or 200 neurons). The scale bar represents 300 counts.

(E) Linear decoding performance (ridge regression, correlation of y velocities) from either latents of masked VAEs (red) or naive VAEs (blue) plotted against the mean uncertainty (i.e., returned standard deviation) of the most informative latents. Shown for different masking levels (light, all observed, to dark, 200 neurons masked), depicted in (D). Linear fits per VAE-model instantiation (seed) reveal a negative correlation between uncertainty prediction and decoding performance for the masked (slopes: -2.8 to -1.5; R-squared: 0.84 to 0.98; 10/10 seeds $p<0.005$), but not the naive (slopes: 0.04 to 0.22; R-squared: 0.11 to 0.87; 6/10 seeds $p>0.05$) , approach. In gray, decoding performance directly from spikes. Note that decoding from spikes has no associated latent uncertainty.

conditional distributions of masked body parts. Across different levels of missingness of neural data during the reaching task, masked VAEs revealed higher downstream decoding performance than naive VAEs. In addition, both masked and naive VAEs showed higher decoding performance from inferred latents than a linear decoder from neural population activity directly, consistent with previous studies.[44,46] Furthermore, we showed how to assess the models' uncertainty estimates, a crucial but often neglected aspect in deep-learning-based dimensionality reduction, and verified that our models learn calibrated distributions. In particular, we demonstrated on the monkey-reach dataset that our approach has the desired property of increasing latent uncertainty when predictions are likely wrong. In order to validate this method, we restricted the analyses to examples, where, in principle, we would have access to all modalities both during training and testing. This, however, is only a requirement for validating the method, not for future applications.

### Generality of modeling conditional distributions in neuroscience and beyond

A key contribution of this work lies in linking conditional distributions of neural and behavioral encoding and decoding to probabilistic approaches for dealing with missing data.[34,39] The generality of this approach opens up various possibilities beyond encoding and decoding, as generating conditional samples and performing inference from partial observations have many applications. In experiments such as the fly-walking example where occlusions or tracking issues occur, our approach enables sampling from distributions over the obscured body key points. In addition, the inherent denoising property of VAEs can correct noisy markers, especially when the observation noise is learned explicitly. More generally, data imputation can be framed as modeling conditional distributions over missing variables given observed ones. This is a relevant pre-processing step for many downstream analyses that require complete datasets in neuroscientific and clinical applications, as well as in other domains.[47,48] For example, if an electrode breaks during a neural recording, a masked VAE approach can salvage the dataset by computing conditionals for the failed electrode using complete data from other sessions.

### Using and assessing uncertainties in deep-learning-based models

It is well established that both modern deep-learning models[49] and traditional Bayesian decoders[50] can make overconfident predictions. In contrast, trustworthy, well-calibrated models should exhibit high uncertainty when predictions are likely to be inaccurate and low uncertainty when they are likely to be accurate. In neuroscientific applications, the ability to assess uncertainty can be particularly important, for example in tasks where wrong predictions can have serious consequences, such as brain-computer interfaces and real-time decoding for an actuator. Classical VAEs provide access to the predicted uncertainty over the inferred latent states, often in the form of the variance of a Gaussian approximate posterior distribution.[32,33] However, this aspect is sometimes treated as a convenience for robust training rather than as a meaningful quantity with respect to the system under investigation. In this work, we have demonstrated the effect of increased latent uncertainty when dealing with partially observed inputs, highlighting the need for modeling of latent uncertainty. Note, however, that correctly capturing uncertainty still poses a challenge for VAEs on many real-world datasets.[51] Further, the observation noise process can be modeled explicitly in VAEs allowing the combination of different data types[34,35]: e.g., Poisson noise for spike counts and Gaussian noise for behavioral trajectories. Using a Gaussian negative log-likelihood loss allows estimation of observation noise for each data channel separately, a desirable property in scientific measurements, yet challenging to accomplish.[52–54] To assess overall calibration in VAEs, which reflects both the posterior uncertainty and observation noise, we introduced a version of simulation-based calibration,[41,42] which allows for sample-based uncertainty evaluation in the absence of tractable ground-truth distributions. Assessing calibration for discrete data poses additional challenges—in particular in a low count regime where it is not possible to obtain reliable confidence intervals—and remains an avenue for future investigation.[55]

### Limitations of the study

Our masked VAE approach allows for calibrated predictions on a variety of conditioning tasks, but it has some limitations. First, our approach relies on a small number of specified and structured conditioning masks, rather than considering all possible combinatorial ($2^D$) masking conditions, where $D$ is the data dimension. For many problems in neuroscience, this suffices since the conditional distributions of interest are usually few and well specified, such as behavior given neural data. To tackle the problem of capturing *all* conditionals, it would be an empirical question of how big models and datasets would need to be to effectively generalize to this combinatorial space. Furthermore, we have only evaluated our approach on data with simultaneous recordings of neurons (and behavior) and have yet to consider the case where, for example, different sets of neurons are recorded at different times. Previous work has demonstrated that it is possible to "stitch" neural population dynamics across multiple populations using non-VAE-based approaches.[56] Extending our method with such approaches is an exciting avenue for future work. Second, similar to other deep-learning-based methods, our approach struggles with capturing interactions on long and varying timescales. Here, we only investigated cross-modality interactions occurring on similar timescales (reach movements) and fixed the length of the time segments during training (maximum of 150 time steps). Thus, behavior or neural activity preceding this segment cannot influence subsequent predictions. Integrating transformer-based approaches[57,58] might be useful for capturing such interactions over varying timescales.[59–62] Third, our approach inherits common issues from VAEs, for example, the lack of a principled way to choose the dimensionality of the VAE latent space, rendering hyperparameter tuning potentially costly. If the dimensionality of the latent space is too large, the VAE might fail to exploit correlations within the data and use separate latent variables for the specified conditional distributions, potentially degrading the quality of conditional generation. On the other hand, if the dimensionality is too small, the VAE might not be able to accurately model the data. Thus,

for the monkey-reach task, we introduced a sparsity-inducing prior[63] that mitigates this issue by automatically reducing the latent dimensionality if latents are not used by the decoder network. Furthermore, deep-learning-based methods such as VAEs can sometimes require large amounts of data. While it is generally difficult to assess *a priori* how much data is required to reach maximum performance, we here show how to get an intuition through varying the training set size (Figure S3). In the Gaussian tractable example, masked VAEs achieve good performance with as few as 100 samples. Fourth, in this paper, we have not focused on the disentanglement of latents. For example, in the monkey encoding and decoding experiments, latents may contain information about both neural activity and behavior. Ideally, we would like to separate neural-only, behavior-only, and shared latents through disentanglement, as proposed in Higgins et al.[64,65] However, theoretical work has shown that, without external supervision, this is challenging.[66] Semi-supervised approaches, as proposed and applied in the neuroscientific context in Whiteway et al. and Yi et al.,[67,68] may be promising to explore in future work. Finally, while disentangled latent spaces is desirable for their interpretability, they do not necessarily lead to models with better generalization, for example, when reconstructing previously unseen combinations of different modalities (e.g., shape and color[69]). Lastly, while samples from our masked VAE are well-calibrated in most cases and often close to the ground-truth neural and behavioral trajectories, the sampling quality of VAEs is known to be limited even for simpler and fully observed datasets. Recent generative models such as Denoising Diffusion Probabilistic Models,[70,71] Normalizing Flows,[72] and Generative Adversarial Networks[73] can produce samples of higher quality, but they lack the main feature of VAEs that makes them especially relevant in neuroscience: inference of low-dimensional latent states. Combining our approach with other such generative models (e.g., Bashiri et al. and Vetter et al.[26,48]) could be an interesting future avenue to improve sampling quality while preserving the possibility of performing latent inference.

## Conclusions

We present a method that addresses two common goals in neuroscience: inferring low-dimensional representations and unveiling dependencies in simultaneously recorded modalities by modeling their conditional distributions. Our approach will allow for scaling encoding and decoding analyses in neuroscience to today's high-dimensional multi-modal datasets. Furthermore, this work highlights a crucial aspect of analyzing neural and behavioral data: the importance of uncertainty estimates.

## RESOURCE AVAILABILITY

### Lead contact

Requests for further information and resources should be directed to and will be fulfilled by the lead contact, Auguste Schulz (auguste.schulz@uni-tuebingen.de).

### Materials availability

This study did not generate any new materials.

### Data and code availability

- The fly-walking dataset has been deposited at https://zenodo.org/records/11002776 and is publicly available as of the date of publication. The DOI is listed in the key resources table.
- The monkey-reach dataset by O'Doherty et al.[40] has been deposited at https://zenodo.org/records/583331 and is publicly available as of the date of publication. The DOI is listed in the key resources table.
- Code for this study and for generating the simulation dataset has been deposited at https://github.com/mackelab/neuro-behavior-conditioning and is publicly available as of the date of publication. The DOI is listed in the key resources table. The code is written in Python; important packages we used include PyTorch,[74] Sklearn,[75] DeepLabCut,[1] and Tracktor.[76]

## ACKNOWLEDGMENTS

We thank Artur Speiser and Paul Fischer for data management and technical support, Lisa Haxel and Michael Deistler for feedback on the manuscript, and all Mackelab members for discussions. This work was supported by the German Research Foundation (DFG) through Germany's Excellence Strategy (EXC-Number 2064/1, PN 390727645) and SFB1233 (PN 276693517), SFB 1089 (PN 227953431), the German Federal Ministry of Education and Research (Tübingen AI Center, FKZ: 01IS18039), and the Human Frontier Science Program (HFSP), and the European Union (ERC, DeepCoMechTome, 101089288). A.S. and J.V. are members of the International Max Planck Research School for Intelligent Systems (IMPRS-IS). D.M. acknowledges a Marie Curie EuroTech postdoctoral fellowship, a Swiss Government Excellence Postdoctoral Scholarship (2018.0483), and funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement no. 754462. V.L.-R. acknowledges support from the Mexican National Council for Science and Technology, CONACYT, under the grant number 709993. P.R. acknowledges support from an SNSF Project grant (no. 175667) and an SNSF Eccellenza grant (no. 181239).

## AUTHOR CONTRIBUTIONS

Conceptualization, A.S., P.R., P.J.G., and J.H.M.; methodology, A.S., D.M., V.L.-R., and P.R.; software, A.S. and V.L.-R.; validation, A.S.; formal analysis, A.S.; investigation, A.S., D.M., P.J.G., and J.H.M.; resources, A.S.; data curation, A.S., D.M., and V.L.-R.; writing – original draft, A.S.; writing – review & editing, A.S., J.V., R.G., D.M., V.L.-R., P.R., P.J.G., and J.H.M.; visualization, A.S.; supervision, P.J.G. and J.H.M.; project administration, A.S. and J.H.M.; funding acquisition, P.R. and J.H.M.

## DECLARATION OF INTERESTS

The authors declare no competing interests.

## DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES

During the preparation of this work, the authors used GitHub Copilot and ChatGPT in order to format LaTeX tables and edit code. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

## STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS
  - Drosophila melanogaster
  - Rhesus macaque monkeys
- METHOD DETAILS
  - Background on variational autoencoders

## SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.celrep.2025.115338.

Received: May 14, 2024
Revised: November 5, 2024
Accepted: January 30, 2025
Published: February 21, 2025

## REFERENCES

1. Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., and Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. Nat. Neurosci. *21*, 1281–1289. https://doi.org/10.1038/s41593-018-0209-y.
2. Günel, S., Rhodin, H., Morales, D., Campagnolo, J., Ramdya, P., and Fua, P. (2019). DeepFly3D, a deep learning-based approach for 3D limb and appendage tracking in tethered, adult Drosophila. Elife *8*, e48571. https://doi.org/10.7554/eLife.48571.
3. Pereira, T.D., Aldarondo, D.E., Willmore, L., Kislin, M., Wang, S.S.H., Murthy, M., and Shaevitz, J.W. (2019). Fast animal pose estimation using deep neural networks. Nat. Methods *16*, 117–125. https://doi.org/10.1038/s41592-018-0234-5.
4. Ahrens, M.B., Orger, M.B., Robson, D.N., Li, J.M., and Keller, P.J. (2013). Whole-brain functional imaging at cellular resolution using light-sheet microscopy. Nat. Methods *10*, 413–420. https://doi.org/10.1038/nmeth.2434.
5. Sofroniew, N.J., Flickinger, D., King, J., and Svoboda, K. (2016). A large field of view two-photon mesoscope with subcellular resolution for in vivo imaging. Elife *5*, e14472. https://doi.org/10.7554/eLife.14472.
6. Jun, J.J., Steinmetz, N.A., Siegle, J.H., Denman, D.J., Bauza, M., Barbarits, B., Lee, A.K., Anastassiou, C.A., Andrei, A., Aydın, Ç., et al. (2017). Fully integrated silicon probes for high-density recording of neural activity. Nature *551*, 232–236. https://doi.org/10.1038/nature24636.
7. de Vries, S.E.J., Lecoq, J.A., Buice, M.A., Groblewski, P.A., Ocker, G.K., Oliver, M., Feng, D., Cain, N., Ledochowitsch, P., Millman, D., et al. (2020). A large-scale standardized physiological survey reveals functional organization of the mouse visual cortex. Nat. Neurosci. *23*, 138–151. https://doi.org/10.1038/s41593-019-0550-9.
8. Siegle, J.H., Jia, X., Durand, S., Gale, S., Bennett, C., Graddis, N., Heller, G., Ramirez, T.K., Choi, H., Luviano, J.A., et al. (2021). Survey of spiking in the mouse visual system reveals functional hierarchy. Nature *592*, 86–92. https://doi.org/10.1038/s41586-020-03171-x.
9. Mimica, B., Tombaz, T., Battistin, C., Fuglstad, J.G., Dunn, B.A., and Whitlock, J.R. (2023). Behavioral decomposition reveals rich encoding structure employed across neocortex in rats. Nat. Commun. *14*, 3947. https://doi.org/10.1038/s41467-023-39520-3.
10. Sani, O.G., Pesaran, B., and Shanechi, M.M. (2021a). Where is all the nonlinearity: flexible nonlinear modeling of behaviorally relevant neural dynamics using recurrent neural networks. preprint at bioRxiv. https://doi.org/10.1101/2021.09.03.458628.
11. Kriegeskorte, N., and Douglas, P.K. (2019). Interpreting encoding and decoding models. Curr. Opin. Neurobiol. *55*, 167–179. https://doi.org/10.1016/j.conb.2019.04.002.
12. Pillow, J.W., Shlens, J., Paninski, L., Sher, A., Litke, A.M., Chichilnisky, E.J., and Simoncelli, E.P. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. Nature *454*, 995–999. https://doi.org/10.1038/nature07140.
13. Paninski, L., and Cunningham, J.P. (2018). Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience. Curr. Opin. Neurobiol. *50*, 232–241. https://doi.org/10.1016/j.conb.2018.04.007.
14. Chen, Z.S., and Pesaran, B. (2021). Improving scalability in systems neuroscience. Neuron *109*, 1776–1790. https://doi.org/10.1016/j.neuron.2021.03.025.
15. Yu, B.M., Cunningham, J.P., Santhanam, G., Ryu, S.I., Shenoy, K.V., and Sahani, M. (2009). Gaussian-Process Factor Analysis for Low-Dimensional Single-Trial Analysis of Neural Population Activity. J. Neurophysiol. *102*, 614–635. https://doi.org/10.1152/jn.90941.2008.
16. Sussillo, D., Jozefowicz, R., Abbott, L.F., and Pandarinath, C. (2016). LFADS - latent factor analysis via dynamical systems. preprint at arXiv. https://doi.org/10.48550/arXiv.1608.06315.
17. Batty, E., Whiteway, M., Saxena, S., Biderman, D., Abe, T., Musall, S., Gillis, W., Markowitz, J., Churchland, A., Cunningham, J.P., et al. (2019). Behavenet: nonlinear embedding and bayesian neural decoding of behavioral videos. In Advances in Neural Information Processing Systems, *32* (Curran Associates, Inc.), pp. 15706–15717.
18. Keeley, S.L., Zoltowski, D.M., Aoi, M.C., and Pillow, J.W. (2020). Modeling statistical dependencies in multi-region spike train data. Curr. Opin. Neurobiol. *65*, 194–202. https://doi.org/10.1016/j.conb.2020.11.005.
19. Sani, O.G., Abbaspourazad, H., Wong, Y.T., Pesaran, B., and Shanechi, M.M. (2021b). Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification. Nat. Neurosci. *24*, 140–149. https://doi.org/10.1038/s41593-020-00733-0.
20. Luxem, K., Mocellin, P., Fuhrmann, F., Kürsch, J., Miller, S.R., Palop, J.J., Remy, S., and Bauer, P. (2022). Identifying behavioral structure from deep variational embeddings of animal motion. Commun. Biol. *5*, 1267. https://doi.org/10.1038/s42003-022-04080-7.
21. Schneider, S., Lee, J.H., and Mathis, M.W. (2023). Learnable latent embeddings for joint behavioural and neural analysis. Nature *617*, 360–368. https://doi.org/10.1038/s41586-023-06031-6.
22. Pfau, D., Pnevmatikakis, E.A., and Paninski, L. (2013). Robust learning of low-dimensional dynamics from large neural ensembles. In Advances in Neural Information Processing Systems, *26* (Curran Associates, Inc.), pp. 2391–2399.
23. Schimel, M., Kao, T.C., Jensen, K.T., and Hennequin, G. (2021). iLQR-VAE : control-based learning of input-driven dynamics with applications to neural data. In International Conference on Learning Representations. https://doi.org/10.1101/2021.10.07.463540.
24. Jensen, K., Kao, T.C., Stone, J., and Hennequin, G. (2021). Scalable Bayesian GPFA with automatic relevance determination and discrete noise models. In Advances in Neural Information Processing Systems, *34* (Curran Associates, Inc.), pp. 10613–10626.
25. Hurwitz, C., Srivastava, A., Xu, K., Jude, J., Perich, M., Miller, L., and Hennig, M. (2021). Targeted neural dynamical modeling. In Advances in Neural Information Processing Systems, *34* (Curran Associates, Inc.), pp. 29379–29392.
26. Bashiri, M., Walker, E., Lurz, K.K., Jagadish, A., Muhammad, T., Ding, Z., Ding, Z., Tolias, A., and Sinz, F. (2021). A flow-based latent state generative model of neural population responses to natural images. In Advances in Neural Information Processing Systems, *34* (Curran Associates, Inc.), pp. 15801–15815.
27. Macke, J.H., Buesing, L., Cunningham, J.P., Yu, B.M., Shenoy, K.V., and Sahani, M. (2011). Empirical models of spiking in neural populations. In Advances in Neural Information Processing Systems, *24* (Curran Associates, Inc.), pp. 1350–1358.
28. Petreska, B., Yu, B.M., Cunningham, J.P., Santhanam, G., Ryu, S., Shenoy, K.V., and Sahani, M. (2011). Dynamical segmentation of single trials from population neural data. In Advances in Neural Information Processing Systems, *24* (Curran Associates, Inc.), pp. 756–764.

29. Linderman, S., Johnson, M., Miller, A., Adams, R., Blei, D., and Paninski, L. (2017). Bayesian Learning and Inference in Recurrent Switching Linear Dynamical Systems. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics Vol. 54 of *Proceedings of Machine Learning Research*, A. Singh and J. Zhu, eds. (PMLR), pp. 914–922.

30. Buesing, L., Macke, J.H., and Sahani, M. (2012). Spectral learning of linear dynamics from generalised-linear observations with application to neural population data. In Advances in Neural Information Processing Systems, *25* (Curran Associates, Inc.), pp. 1691–1699.

31. Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T., and Alameda-Pineda, X. (2021). Dynamical Variational Autoencoders: A Comprehensive Review. FNT. in Machine Learning *15*, 1–175. https://doi.org/10.1561/2200000089.

32. Rezende, D.J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In Proceedings of the 31st International Conference on Machine Learning Vol. 32 of *Proceedings of Machine Learning Research*, E.P. Xing and T. Jebara, eds. (Bejing, China: PMLR), pp. 1278–1286.

33. Kingma, D.P., and Welling, M. (2014). Auto-encoding variational Bayes. In International Conference on Learning Representations. https://doi.org/10.48550/arXiv.1312.6114.

34. Nazábal, A., Olmos, P.M., Ghahramani, Z., and Valera, I. (2020). Handling incomplete heterogeneous data using VAEs. Pattern Recogn. *107*, 107501. https://doi.org/10.1016/j.patcog.2020.107501.

35. Brenner, M., Hess, F., Koppe, G., and Durstewitz, D. (2024). Integrating multimodal data for joint generative modeling of complex dynamics. arXiv *235*, 4482–4516.

36. Zhou, D., and Wei, X.X. (2020). Learning identifiable and interpretable latent models of high-dimensional neural activity using pi-VAE. In Advances in Neural Information Processing Systems, *33* (Curran Associates, Inc.), pp. 7234–7247.

37. Williams, C.K.I., Nash, C., and Nazábal, A. (2019). Autoencoders and Probabilistic Inference with Missing Data: An Exact Solution for The Factor Analysis Case. preprint at arXiv. https://doi.org/10.48550/arXiv.1801.03851.

38. Ivanov, O., Figurnov, M., and Vetrov, D. (2019). Variational autoencoder with arbitrary conditioning. In International Conference on Learning Representations. arXiv:1806.02382.

39. Collier, M., Nazábal, A., and Williams, C.K.I. (2020). VAEs in the presence of missing data. In ICML Workshop on the Art of Learning with Missing Values (Artemiss). arXiv:2006.05301.

40. O'Doherty, J.E., Cardoso, M.M.B., Makin, J.G., and Sabes, P.N. (2017). Nonhuman Primate Reaching with Multichannel Sensorimotor Cortex Electrophysiology (Zenodo). https://doi.org/10.5281/zenodo.583331.

41. Cook, S.R., Gelman, A., and Rubin, D.B. (2006). Validation of Software for Bayesian Models Using Posterior Quantiles. J. Comput. Graph Stat. *15*, 675–692. https://doi.org/10.1198/106186006X136976.

42. Talts, S., Betancourt, M., Simpson, D., Vehtari, A., and Gelman, A. (2018). Validating Bayesian Inference Algorithms with Simulation-Based Calibration.. preprint at arXiv. https://doi.org/10.48550/arXiv.1804.06788.

43. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), A. Moschitti, B. Pang, and W. Daelemans, eds. (Doha, Qatar: Association for Computational Linguistics), pp. 1724–1734. https://doi.org/10.3115/v1/D14-1179.

44. Makin, J.G., O'Doherty, J.E., Cardoso, M.M.B., and Sabes, P.N. (2018). Superior arm-movement decoding from cortex with a new, unsupervised-learning algorithm. J. Neural. Eng. *15*, 026010. https://doi.org/10.1088/1741-2552/aa9e95.

45. Pei, F.C., Ye, J., Zoltowski, D.M., Wu, A., Chowdhury, R.H., Sohn, H., O'Doherty, J.E., Shenoy, K.V., Kaufman, M., Churchland, M.M., et al. (2021). Neural latents benchmark '21: Evaluating latent variable models of neural population activity. In Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2). arXiv:2109.04463.

46. Afshar, A., Santhanam, G., Yu, B.M., Ryu, S.I., Sahani, M., and Shenoy, K.V. (2011). Single-trial neural correlates of arm movement preparation. Neuron *71*, 555–564. https://doi.org/10.1016/j.neuron.2011.05.047.

47. Talukder, S.J., Sun, J.J., Leonard, M.K., Brunton, B.W., and Yue, Y. (2022). Deep neural imputation: A framework for recovering incomplete brain recordings. In NeurIPS 2022 Workshop on Learning from Time Series for Health. arXiv:2206.08094v1.

48. Vetter, J., Macke, J.H., and Gao, R. (2024). Generating realistic neurophysiological time series with denoising diffusion probabilistic models. Patterns *5*, 101047. https://doi.org/10.1016/j.patter.2024.101047.

49. Guo, C., Pleiss, G., Sun, Y., and Weinberger, K.Q. (2017). On calibration of modern neural networks. In Proceedings of the 34th International Conference on Machine Learning Vol. 70 of *Proceedings of Machine Learning Research*, D. Precup and Y.W. Teh, eds. (PMLR), pp. 1321–1330.

50. Wei, G., Mansouri, Z.T., Wang, X., and Stevenson, I.H. (2024). Calibrating Bayesian decoders of neural spiking activity. Journal of Neuroscience *44*, e2158232024. https://doi.org/10.1523/JNEUROSCI.2158-23.2024.

51. Wang, Y., Blei, D., and Cunningham, J.P. (2021). Posterior collapse and latent variable non-identifiability. In Advances in Neural Information Processing Systems, *34* (Curran Associates, Inc.), pp. 5443–5455.

52. Rybkin, O., Daniilidis, K., and Levine, S. (2021). Simple and Effective VAE Training with Calibrated Decoders. In Proceedings of the 38th International Conference on Machine Learning Vol. 139 of *Proceedings of Machine Learning Research*, M. Meila and T. Zhang, eds. (PMLR), pp. 9179–9189.

53. Skafte, N., Jørgensen, M., and Hauberg, S. (2019). Reliable training and estimation of variance networks. In Advances in Neural Information Processing Systems, *32* (Curran Associates, Inc).

54. Seitzer, M., Tavakoli, A., Antic, D., and Martius, G. (2022). On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks. In International Conference on Learning Representations. https://doi.org/10.48550/arXiv.2203.09168.

55. Wei, W., and Held, L. (2014). Calibration tests for count data. TEST *23*, 787–805. https://doi.org/10.1007/s11749-014-0380-8.

56. Turaga, S., Buesing, L., Packer, A.M., Dalgleish, H., Pettit, N., Hausser, M., and Macke, J.H. (2013). Inferring neural population dynamics from multiple partial recordings of the same neural circuit. In Advances in Neural Information Processing Systems (Curran Associates, Inc.).

57. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, .u., and Polosukhin, I. (2017). Attention is All you Need. In Advances in Neural Information Processing Systems, *30* (Curran Associates, Inc.). arXiv:1706.03762.

58. Jaegle, A., Borgeaud, S., Alayrac, J.B., Doersch, C., Ionescu, C., Ding, D., Koppula, S., Zoran, D., Brock, A., Shelhamer, E., et al. (2022). Perceiver IO: A general architecture for structured inputs & outputs. In International Conference on Learning Representations. arXiv:2107.14795.

59. Ye, J., and Pandarinath, C. (2021). Representation learning for neural population activity with neural data transformers. Neuron. Behav. Data Anal. Theory *5*, 1–18. https://doi.org/10.51628/001c.27358.

60. Ye, J., Collinger, J., Wehbe, L., and Gaunt, R. (2023). Neural Data Transformer 2: Multi-context Pretraining for Neural Spiking Activity. In Advances in Neural Information Processing Systems (Curran Associates, Inc.), pp. 80352–80374.

61. Azabou, M., Arora, V., Ganesh, V., Mao, X., Nachimuthu, S., Mendelson, M., Richards, B., Perich, M., Lajoie, G., and Dyer, E. (2023). A Unified, Scalable Framework for Neural Population Decoding. In Advances in Neural Information Processing Systems (Curran Associates, Inc.), pp. 44937–44956.

62. Antoniades, A., Yu, Y., Canzano, J.S., Wang, W.Y., and Smith, S. (2024). Neuroformer: Multimodal and Multitask Generative Pretraining for Brain Data. In International Conference on Learning Representations. arXiv:2311.00136.

63. Ainsworth, S.K., Foti, N.J., Lee, A.K.C., and Fox, E.B. (2018). oi-VAE: Output interpretable VAEs for nonlinear group factor analysis. In Proceedings of the 35th International Conference on Machine Learning, J. Dy and A. Krause, eds. (PMLR), pp. 119–128.

64. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In International Conference on Learning Representations.

65. Higgins, I., Chang, L., Langston, V., Hassabis, D., Summerfield, C., Tsao, D., and Botvinick, M. (2021). Unsupervised deep learning identifies semantic disentanglement in single inferotemporal face patch neurons. Nat. Commun. *12*, 6456. https://doi.org/10.1038/s41467-021-26751-5.

66. Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In Proceedings of the 36th International Conference on Machine Learning, K. Chaudhuri and R. Salakhutdinov, eds. (PMLR), pp. 4114–4124.

67. Whiteway, M.R., Biderman, D., Friedman, Y., Dipoppa, M., Buchanan, E.K., Wu, A., Zhou, J., Bonacchi, N., Miska, N.J., Noel, J.P., et al. (2021). Partitioning variability in animal behavioral videos using semi-supervised variational autoencoders. PLoS Comput. Biol. *17*, e1009439. https://doi.org/10.1371/journal.pcbi.1009439.

68. Yi, D., Musall, S., Churchland, A., Padilla-Coreano, N., and Saxena, S. (2023). Disentangled multi-subject and social behavioral representations through a constrained subspace variational autoencoder (cs-vae). Elife *12*, e88602. https://doi.org/10.7554/eLife.88602.1.

69. Montero, M.L., Ludwig, C.J., Costa, R.P., Malhotra, G., and Bowers, J. (2020). The role of disentanglement in generalisation. In International Conference on Learning Representations.

70. Ho, J., Jain, A., and Abbeel, P. (2020). Denoising Diffusion Probabilistic Models. In Advances in Neural Information Processing Systems, *33* (Curran Associates, Inc.), pp. 6840–6851.

71. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 10684–10695.

72. Rezende, D.J., and Mohamed, S. (2015). In Variational inference with normalizing flows, *Vol.* (Lille, France: 37 of Proceedings of Machine Learning Research), pp. 1530–1538.

73. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Advances in Neural Information Processing Systems, *27* (Curran Associates, Inc.). arXiv:1406.2661.

74. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems (Curran Associates, Inc.).

75. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. J. Mach. Learn. Res. *12*, 2825–2830.

76. Sridhar, V.H., Roche, D.G., and Gingins, S. (2019). Tracktor: Image-based automated tracking of animal movement and behaviour. Methods Ecol. Evol. *10*, 815–820. https://doi.org/10.1111/2041-210X.13166.

77. Sen, R., Wu, M., Branson, K., Robie, A., Rubin, G.M., and Dickson, B.J. (2017). Moonwalker descending neurons mediate visually evoked retreat in drosophila. Curr. Biol. *27*, 766–771. https://doi.org/10.1016/j.cub.2017.02.008.

78. Kingma, D.P., and Welling, M. (2019). An introduction to variational autoencoders. FNT. in Machine Learning *12*, 307–392. https://doi.org/10.1561/2200000056.

79. Gregor, K., Danihelka, I., Graves, A., Rezende, D., and Wierstra, D. (2015). Draw: A recurrent neural network for image generation. In Proceedings of the 32nd International Conference on Machine Learning Vol. 37 of *Proceedings of Machine Learning Research*, F. Bach and D. Blei, eds. (Lille, France: PMLR), pp. 1462–1471.

80. Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A., and Bengio, Y. (2016). A Recurrent Latent Variable Model for Sequential Data. In Proceedings of the 28th International Conference on Neural Information Processing Systems -, *2*, pp. 2980–2988.

81. Pandarinath, C., O'Shea, D.J., Collins, J., Jozefowicz, R., Stavisky, S.D., Kao, J.C., Trautmann, E.M., Kaufman, M.T., Ryu, S.I., Hochberg, L.R., et al. (2018). Inferring single-trial neural population dynamics using sequential auto-encoders. Nat. Methods *15*, 805–815.

82. Savitzky, A., and Golay, M.J.E. (1964). Smoothing and differentiation of data by simplified least squares procedures. Anal. Chem. *36*, 1627–1639. https://doi.org/10.1021/ac60214a047.

83. Kingma, D.P., and Ba, J. (2015). Adam: A method for stochastic optimization. In International Conference on Learning Representations. arXiv:1412.6980.

84. Loshchilov, I., and Hutter, F. (2019). Decoupled Weight Decay Regularization. In International Conference on Learning Representations. arXiv:1711.05101.

85. Javaloy, A., Meghdadi, M., and Valera, I. (2022). Mitigating modality collapse in multimodal VAEs via impartial optimization. In Proceedings of the 39th International Conference on Machine Learning Vol. 162 of *Proceedings of Machine Learning Research*, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, eds. (PMLR), pp. 9938–9964.

86. Biewald, L. (2020). Experiment tracking with weights and biases. URL: https://www.wandb.com/softwareavailablefromwandb.com.

## STAR★METHODS

### KEY RESOURCES TABLE

| REAGENT or RESOURCE | SOURCE | IDENTIFIER |
|---|---|---|
| Deposited data | | |
| Walking behavior of Drosophila melanogaster | this paper | https://doi.org/10.5281/zenodo.11002775 |
| Experimental models: Organisms/strains | | |
| *D. melanogaster: Moonwalker fly: 20XUAS-CsChrimson-mVenus (attP18)/+; 050660-p65ADZp (attP40)/+; 044845-ZpGAL4DBD (attP2)/+* | Sen et al.[77] | MDN-3 |
| Public Monkey Reach Dataset | O'Doherty et al.[40] | https://doi.org/10.5281/zenodo.788569 |
| Software and algorithms | | |
| Original code | this paper | https://github.com/mackelab/neuro-behavior-conditioning https://doi.org/10.5281/zenodo.14766113 |
| Pytorch | Paszke et al.[74] | https://github.com/pytorch/pytorch |
| Sklearn | Pedregosa et al.[75] | https://github.com/scikit-learn/scikit-learn |
| Python | Python Software Foundation | https://www.python.org/ |
| DeepLabCut | Mathis et al.[1] | http://www.mackenziemathislab.org/deeplabcut |
| Tracktor | Sridhar et al.[76] | https://github.com/vivekhsridhar/tracktor |

### EXPERIMENTAL MODEL AND STUDY PARTICIPANT DETAILS

#### Drosophila melanogaster

Female flies (Drosophila melanogaster) were placed in an acrylic arena that constrained them to move in a 2D plane. The flies were part of a genetic screen (20XUAS-CsChrimson-mVenus (attP18)/+; 050660-p65ADZp (attP40)/+; 044845-ZpGAL4DBD (attP2)/+,[77] i.e., not wild-type), but were largely morphologically and behaviorally indistinguishable from wild-type flies. All experiments were performed with adult flies, i.e., at least 7 days after emerging from their pupa. Flies were maintained at 25°C and 50% humidity. All experiments were performed in compliance with relevant national (Switzerland) and institutional (EPFL) ethical regulations.

#### Rhesus macaque monkeys

Neural and behavioral data from a male Rhesus macaque monkey (Macaca mulatta) who performed self-paced reaches were recorded and made publicly available by O'Doherty et al.[40] At the time of data collection, the monkey was 9 years old and weighed 14.5 kg.[44] As stated by the lab performing the experiments, all animal procedures were performed in accordance with the U.S. National Research Council's Guide for the Care and Use of Laboratory Animals and were approved by the UCSF Institutional Animal Care and Use Committee.[40,44]

Given the methodological nature, we do not expect any sex related influence on our results.

### METHOD DETAILS

Here, we adapt variational autoencoders[32,33] to address two goals simultaneously: First, to infer low-dimensional representations underlying multi-modal neural and behavioral time-series data and, second, to model their conditional distributions. Modeling conditional distributions is ubiquitous in neuroscience, and since neuroscientific data are typically variable even in controlled experiments, relations between modalities may also be variable. Therefore, we focus on probabilistic rather than deterministic approaches to characterize such conditional distributions. We reformulate the estimation of conditional distributions in VAEs in a more general way: modeling the distribution of an unobserved subset of the data given an observed subset $p(\text{unobserved}|\text{observed})$ similar to Nazábal et al. and Collier et al.[34,39] To target such distributions with a VAE, we modify the training scheme and loss of classical VAEs. We validate our approach on a tractable example and two neuroscientific time-series datasets: walking behavior of the fly and a continuous reach task in monkeys. We introduce calibration metrics to evaluate the models' uncertainty estimates in the context of scientific data, i.e., without access to ground-truth uncertainties.

## Background on variational autoencoders

Variational Autoencoders (VAEs) are probabilistic models capable of capturing complex multi-modal data distributions $p(\mathbf{x})$. The assumption underlying VAEs is that all variations in the data distributions can be captured (up to observation/measurement noise) by the variations of corresponding unobserved latent variables $z$. VAEs learn stochastic mappings between the observed data space and the unobserved or latent ($z$-space). Both mappings from the data to latent distributions and vice versa are typically parameterized through flexible neural networks. The generative model is described by the joint distribution of data and latent variables, which factorizes into

$$p_\theta(\mathbf{x},\mathbf{z}) = p_\theta(\mathbf{z})p_\theta(\mathbf{x}|\mathbf{z}) \tag{1}$$

parameterized by $\theta$ where $p(\mathbf{z})$ is the prior over the latent space. The prior is usually chosen to be a simple distribution such as a standard Gaussian, and $p_\theta(\mathbf{x}|\mathbf{z})$ is the probabilistic decoder. The inference or encoder model $q_\phi(\mathbf{z}|\mathbf{x})$, parameterized by $\phi$, which infers the latent distribution from data, is an approximation of the true, intractable posterior $p(\mathbf{z}|\mathbf{x})$[33,32,78]. VAEs are trained by maximizing a lower bound of the data log-likelihood. This so-called Evidence Lower Bound (ELBO) can be written as:

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathrm{D}_{\mathrm{KL}}\left[q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right]. \tag{2}$$

The first term assesses how well the predicted distribution matches the original data and is often referred to as the reconstruction loss. The second term is the Kullback-Leibler divergence $\mathrm{D}_{\mathrm{KL}}$ between the approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ and the latent prior $p(\mathbf{z})$, which regularizes the learned latent space. Maximizing the ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ with respect to the parameters $\theta$ and $\phi$ leads to a better generative model and increases the similarity between the approximate and the true (intractable) posterior. All parameters $\theta$ and $\phi$ can be optimized jointly using stochastic gradient descent. Once trained successfully, one can sample from the prior and consecutively from the stochastic decoder output to obtain a new sample $\hat{x}_{\mathrm{pred}}$ from the learned data distribution. Alternatively, one can sample from the approximate posterior of a previously unseen test datum $x_{\mathrm{test}}$ to obtain reconstructions that closely resemble $x_{\mathrm{test}}$. VAEs have been successfully applied to various types of potentially heterogeneous data (continuous, discrete, ordinal, etc.)[34] and have been extended to time series,[31,79,80] paving the way for applications to neuroscientific time series.[16,20,35,81]

## Capturing arbitrary conditional distributions with VAEs

To model flexible conditional distributions with VAEs, we modified the training scheme of classical VAEs similar to Nazábal et al. and Collier et al.[34,39] While training the VAE, we randomly mask out subsets of the data corresponding to the desired conditional distributions and compute the loss on the remaining data (see Method S1). Selecting a subset of the data that is held out for specific analyses may seem related to cross-validation. It is important to distinguish that while cross-validation separates independent subsets of the data to assess the generalization properties of a model, here, the data selection happens in terms of individual features, i.e., dependent data dimensions. Prior to training, for each conditional distribution of interest, we specify a conditioning mask $m$ together with a mask probability $p_m$ (Figure 1C, left). During training, the conditioning masks are sampled independently for each data point according to the mask probabilities. Concretely, the masking is performed by replacing the data with their respective mean values. Other replacement values, such as zeros for spiking (count) data, are also possible. We calculate the reconstruction loss $\mathcal{L}_{recon}$ solely on observed, that is, non-masked data. Sometimes, to facilitate learning that some data has been masked out, we additionally provide the encoder network with a binary mask consisting of 0s for unobserved and 1s for observed data points (see Methods, networks). Through this training procedure, the masked VAE simultaneously optimizes the ELBO over all different conditional distributions, that is

$$\mathcal{L}^{\mathrm{masked}}_{\theta,\phi}(\mathbf{x}) = \mathbb{E}_{m\sim p(m)}\left[\mathcal{L}^{m}_{\theta,\phi}(\mathbf{x})\right], \tag{3}$$

where $p(m)$ is the previously specified probability distribution over all conditioning masks, including the fully observed case, where no data is masked out. As noted above, the mask $m$ is applied to both the data and the corresponding part of the reconstruction loss. This training procedure promotes the learning of different encoder networks that share parameters, allowing us to target different approximate posterior distributions given different conditioning masks. From an implementation perspective, the conditioning masks can be passed to the encoder network in various ways. They can, for example, be concatenated or added directly to the input, but also at later stages in the network, possibly after transformations with a (non-linear) embedding. In contrast to Collier et al.,[39] we explicitly do not pass the conditioning masks to the decoder since all uncertainty and mean shifts induced by masking should be reflected in the latent representation.

## Modeling observation noise with VAEs

It is important to ensure that VAEs correctly capture the uncertainties in the (conditional) data distributions. In a VAE, there are two sources of uncertainty - the inferred posterior uncertainty and the observation or measurement noise. The latter source of uncertainty is often ignored, which is reflected in the common choice of the mean squared error (MSE) as the reconstruction loss. The MSE only evaluates the quality of the mean prediction and ignores the stochastic nature of the VAE decoder. If we instead want to correctly capture the observation noise, it is necessary to learn it explicitly. Assuming that the observation noise follows a

Gaussian distribution, we use the Gaussian negative log-likelihood (GNLL) as our reconstruction loss. The GNLL for an observation $x$ given a model prediction of the Gaussian mean $\mu$ and standard deviation $\sigma$ is given by

$$\mathcal{L}_{\text{GNLL}}(x;\mu,\sigma) = -\log P(X = x;\mu,\sigma) = \frac{1}{2}\log\left(2\pi\sigma^2\right) + \frac{(x-\mu)^2}{2\sigma^2}. \quad (4)$$

Note that the MSE is a special case of the GNLL where the standard deviation is set to 1. As noted above, this usually leads to samples from the model that are not calibrated in the statistical sense. Optimization, however, is more challenging when using the GNLL and might require additional adjustments.[52,54]

## Datasets and data preprocessing

### Linear Gaussian Latent Variable Model

We simulated a dataset based on a Gaussian Latent Variable Model (GLVM) with one latent variable $z$, where $z \backsim \mathcal{N}(0,1)$, and 20 data dimensions $\mathbf{x}$, where $\mathbf{x} \backsim \mathcal{N}(\mathbf{C}z + \mathbf{d}, \mathbf{\Lambda})$ (Figure 2A). To demonstrate the difference between noisy and more precise, less noisy, variables in a setup that accounts for uncertainty, noise levels for all data dimensions differ. For each data dimension $i$, $\sigma_i$ is drawn from a log-normal distribution with $\mu_{LN} = \log(0.7)$ and $\sigma_{LN} = 0.5$. $C_i$s are drawn from a normal distribution with $\mu_N = 1.1$ and $\sigma_N^2 = 0.1$. Additionally, the sign for a given $C_i$ is flipped with probability 0.5. All offsets $\mathbf{d}$ are set to 1. We use 9000 samples from this model for training and 1000 each for validation and testing. This fully Gaussian setup allows for the analytical computation of both conditional $p(\mathbf{x}^{\text{unobs}}|\mathbf{x}^{\text{obs}})$ and posterior $p(z|\mathbf{x}^{\text{obs}})$ distributions (see Method S2), which we compare to the distributions learned by our model.

### Fly walking behavior

To collect the data on fly walking behavior, we placed flies, *Drosophila melanogaster*, in an acrylic arena that constrained them to move in a 2D plane. Thus, flying is not part of the otherwise rich repertoire of observed behaviors, which includes both forward and backward walking, grooming, resting, etc. The flies were part of a genetic screen (not wild-type) but were examined during behavior capture and were morphologically and behaviorally indistinguishable from wild-type flies. We placed three female flies in one arena simultaneously and filmed them from below, three times for 3 s (frame rate of 80*Hz*). This procedure was repeated about 10000 times, resulting in 28059 time series with 234 time points each. To extract each fly from the video separately, we tracked the centroid of each fly using Tracktor,[76] cropped out the flies in each frame, and aligned them to point in the upward direction. We then tracked 32 body parts (four joints per leg, as well as head features, thorax, abdomen, and wings), each with x- and y-directions using DeepLabCut,[1] resulting in time series with 64 feature dimensions. We then smoothed the extracted time series using a Savitzky-Golay-Filter[82] with a polynomial order of two and a window length of seven. The smoothed trajectories were then cut into sequences of length 48 with buffers of length 9 between each sequence to avoid information leakage. 95% of the data was used for training, while the remaining data was used for validation and testing (2806 sequences each). Potential information leakage due to autocorrelation between training and test/validation sets is further reduced by choosing the last sequences for testing/validation instead of an interleaved approach, which can often cause information leakage in time-series models. Prior to passing the time series to the network, we standardize each feature dimension across the 48 time steps.

### Continuous reach task in monkeys

The neural and behavioral dataset, made publicly available by O'Doherty et al.,[40] was recorded from two monkeys (rhesus macaque) performing self-paced continuous reaches, i.e., without gaps or pre-movement delay intervals. Targets were arranged in an 8 by 8 grid, and a new target was presented when the previous target was reached. Neural recordings were taken from the cortical hemisphere contralateral to the arm performing the reach movements. O'Doherty et al.[40] provide the neural data after spike sorting in the shape channels vs. spike times. We focus only on one session, 'loco_20170213_02', which contains neural activity from both primary motor (M1) and (S1) activity, as well as the cursor, target, and finger positions. Here, we take only the neural activity from M1 and the cursor positions ($x$,$y$ direction) as the behavioral correlate. We filter out channels with firing rates below 0.5 Hz analogous to Makin et al.,[44] resulting in 213 remaining M1 units. We convert spike times into spike counts in bins of 64 ms (15.625 Hz). We down-sample the cursor position by querying it at fewer time points consistent with the reduced sampling rate used for binning the spikes (15.625 Hz instead of 250 Hz). We do not introduce a delay between neural activity and behavior as done, e.g., in Schimel et al. and Jensen et al.,[23,24] we rather let the model identify which aspects of the respective other time series to consider for its predictions. We use the first 70% of the data for training (approx. 28 min recording time), the following 10% for testing (approx. 4 min), and the remaining 20% for validation (approx. 8 min). We standardize the behavioral train, test, and validation time series with respect to the overall mean and standard deviation of the training set for both reach directions. During training, we introduce 'pseudo-trials' with 150 time steps each, that start at randomly sampled time points.

## Network architectures and optimization

### Model details: GLVM

**Training scheme and masks:** Prior to training, we select three randomly sampled masks (10 out of 20 dimensions are masked) to test if the masked approach can capture the true posterior and conditional distributions. Since we chose different loading factors $C_i$ and noise levels $\sigma_i$ for each dimension, the corresponding posterior mean and variance, and thus also the conditional distributions,

differ between conditions. During training, we uniformly sampled the four conditioning masks (all observed and mask 1–3) and used the Adam optimizer[83] to train our model.

**Architecture:** The encoder network consists of a simple linear embedding for the mask, which is passed through a multilayer perceptron together with the 20-dimensional data vector to parameterize the one-dimensional posterior mean and log variance. To focus on posterior inference under different masking conditions, we set the decoder to be the true generative model. Note, however, that this GLVM example is identifiable, i.e., all parameters of the generative model ($C_i, d_i, \sigma_i$) can be learned using a VAE, which we confirmed even in our masked training scheme. Nevertheless, fixing the decoder is beneficial in this case since the posterior is only identifiable up to a rotation in the latent space ($\mu_z, \sigma_z$), which in the one-dimensional setting corresponds to a flipped sign. See Table S1 in the supplement for hyperparameters.

**Loss:** For this well-specified, identifiable Gaussian example, the regular masked GNLL was used together with a standard Gaussian prior in the latent space.

### *Model details: Fly walking behavior*

**Training scheme and masks:** To investigate low-dimensional representations of fly walking behavior, we built a sequential VAE and specified masks for the body keypoints most relevant to walking. Analogous to the GLVM case, we adapted the masked training scheme for the time-series case to allow for modeling the conditional distribution over a subset of the fly body keypoints, given the remaining keypoints. Specifically, we mask the hind claw, hind tibia-tarsal joint, mid tibia-tarsal joint, and mid claw of the left side. The entire time segment of masked keypoints is replaced with the mean value across this segment. We assign a probability of 50% to the all-observed and the leg-masking condition. We again use the Adam optimizer[83] with a learning rate of 0.0005 for training our model.

**Architecture:** The VAE for fly walking behavior consists of an encoder and a decoder network that are both trainable neural networks. The encoder network consists of two sets of 1D convolutional layers, each followed by batch normalization and ELU activation. We then apply temporal convolutions that compress the data in the temporal dimension before passing it to a bidirectional RNN[43] for temporal context. Thus, the encoder network is non-causal in time. The RNN output is then passed through a multi-layer perceptron to parameterize the posterior mean and log variance. This results in a latent space with spatial ($N_z$) and temporal ($T_z$) dimensions smaller than the 64 features and 48 time-steps of the data (for our choice of parameters $N_z = 18$ and $T_z = 13$, i.e., the size of the latent space is less than 8% of the original data). Unlike in the GVLM case, we do not pass the mask to the encoder network, since it does not improve the conditional modeling. After sampling from the approximate posterior, the decoder network expands the time dimension of $\mathbf{z}$ using transposed convolutions, followed by dimensionality expansion to parameterize the Gaussian mean and observation noise variance. The latter is constrained to be positive by a softplus function to ensure well-defined variances. See Table S2 in the supplement for hyperparameters.

**Loss:** For the continuous behavioral data, we again use GNLL (Equation 4), which is computed per feature and timepoint. The prior distribution in the latent space is standard Gaussian.

### *Model details: Neural and behavioral data from a monkey reach task*

**Training scheme and masks:** The sequential VAE for the monkey reach task jointly models time series of high-dimensional neural spike-count data and continuous cursor positions. We specify the masks required for neuroscientific encoding and decoding: either all neural activity is masked out (spike counts set to zero), or all behavioral traces are masked and set to their respective mean values. Following Ainsworth et al.,[63] we introduced a sparsity-inducing prior that sets latent contributions to zero if they are not used by the model. We used the AdamW optimizer[84] with a learning rate of 0.001 and weight decay of 0.2) to train our encoder and decoder networks. For parameters related to sparsity-induction, we follow Ainsworth et al.[63] and use Stochastic Gradient Descent (SGD) with zero momentum. Here, we show results that are trained on only one session, but the architecture allows training on data from multiple sessions using session-specific input and output mappings.

**Architecture:** First, we expand the data using a session-specific linear mapping. Similar to the sequential VAE for the fly data, the encoder network then performs a non-linear dimensionality reduction followed by a bidirectional RNN to parameterize the latent posterior mean and log-variance. Here, we do not consider compression in time, and each latent time-point corresponds to a time-point in dataspace. The decoder also has an RNN and uses further multi-layer-perceptrons to map the latent samples back into data space. For the continuous behavioral data, the decoder again predicts the Gaussian mean and observation noise variance. For the discrete spike data, however, the decoder only models the underlying firing rates. See Table S3 in the supplement for hyperparameters.

**Loss:** This discrepancy arises from the different distributions used to model the respective data modalities. While behavior is continuous and thus appropriately modeled with a Gaussian, discrete spike counts are best modeled with a Poisson distribution. Consequently, the GNLL is replaced by the negative Poisson log-likelihood, which, for an observed spike count $x$ and a rate parameter $\lambda$, is defined as:

$$\mathcal{L}_{\text{Poisson}}(x;\lambda) = -\log P(X = x;\lambda) = -x\log(\lambda) + \lambda + \log(x!). \quad (5)$$

For this task, we also refined the behavioral GNLL loss by incorporating the concept of $\beta$-NLL introduced by Seitzer et al.[54] by weighting each data point's contribution to the loss based on the $\beta$-NLL-exponentiated variance estimate $\sigma_i^{2\beta_{NLL}}$. Effectively, this small loss modification prevents a potential issue when learning the observation noise, namely that poorly fitted variables are assigned high variance, which, since it appears in the denominator in Equation 4, leads to smaller gradients and hence less incentives for the network to improve its fit. In this application, an NLL-beta of 0.3 worked well. Before calculating the overall gradients, we sum up

the behavioral and neural contributions to the reconstruction loss. Note that when using heterogeneous noise models the scales of the loss contributions can vastly differ. Hence, depending on the application and downstream tasks, it can be beneficial or even necessary to introduce weighting factors to balance out the losses and corresponding gradients.[85] To improve stability and prevent over-fitting, we additionally regularize the session input and output weight matrices. For the sparsity-inducing prior in the latent space, we apply a version of Lasso regularization that encourages sparsity in the weight matrix that transforms the $z$-samples before they are passed through the decoder (see[63] for details).

**Linear decoding from neural latents:** For the experiment of partial neural observations, we set up identical VAEs with the main difference that the VAE receives no behavioral input and output. We vary between passing in all neural activity and masked activity with 5, 20, 50, 100, 150, or 200 masked neurons of 213. Note that the (Poisson-)loss is then only computed on the remaining observed neurons. Since we are only dealing with a single discrete modality, we did not incorporate the GNLL loss nor any NLL-beta scaling. All other training configurations and parameter settings remained identical. At test time, we apply the masks to the training data and assess the inferred latent means and associated uncertainty estimates (standard deviation returned by the VAE). For each method and model instantiation, we identify the most informative latents as those with the highest variability in the inferred latent mean over time. Unused or less relevant latents converge to the prior, which in these experiments was set to mean 0 and standard deviation 1. Unused latents hardly vary over time and can thus be easily identified post hoc. Setting the correct latent dimension *a priori* poses a challenge (see Limitations). After training the VAE, we trained a linear ridge regression model ($\alpha = 0.01$) to predict behavior (velocity) given latents (means) of the fully observed condition. When training the decoder, we shifted the velocity by 128 ms following Jensen et al.[24] and Schimel et al.[23] We scaled both neural activity and behavior using min-max scaling, with scaling factors obtained on the training set and applied to the validation and test sets. We quantify decoding performance as the correlation between the predicted velocity (y-direction) and the true velocity. We apply the same decoding strategy to raw spiking data, training on all observed spiking activity and predicting velocity from only partial test recordings.

### *Computational cost*

We used Weights and Biases[86] to track VAE training times. Since the architectures of masked and naive VAEs are almost identical and masked VAEs have only a few extra parameters to optimize, the compute times are comparable: For the GLVM task, training naive VAEs took 4.58 ± 0.08 min (mean ± standard deviation) and masked VAEs 4.57 ± 0.10 min; for the fly task, 145.18 ± 24.21 min (naive) and 155.05 ± 15.46 min (masked); for the monkey encoding and decoding task, 7.61 ± 0.12 min (naive) and 7.36 ± 0.06 min (masked); for the monkey neural latents task, 6.22 ± 0.25 min (naive) and 6.21 ± 0.29 min (masked). We did not perform early stopping, so the reported times do not reflect the time until convergence. Furthermore, for naive VAEs, smaller architectures may have been sufficient to accurately model the fully observed case. We trained and evaluated masked and naive VAEs on the Gaussian and Monkey reach datasets on an NVIDIA RTX 3090 (24GB RAM). The Fly dataset, which is significantly larger, was trained on a GeForce RTX 2080 Ti (11GB RAM), which is slower than the 3090. Therefore, the numbers between the different datasets are not directly comparable.

## QUANTIFICATION AND STATISTICAL ANALYSIS

### Calibration metrics: Evaluating uncertainties in variational autoencoders

To investigate the statistical calibration in VAEs, we perform a version of simulation-based calibration,[41,42] which is associated to frequentist coverage tests.[50] Here, we focus on the calibration of the predictive distribution in data space. For each test datum $x_{\text{test}}$, we sample $n_z$-times from the approximate posterior, pass the sampled $z$ through the decoder and sample from the observation noise model $n_{obs}$-times. This sampling procedure results in a sampling distribution in data space that reflects both posterior uncertainty and observation noise.

**For continuous data**, we then compute confidence intervals corresponding to the $n$th percentile. For a statistically well-calibrated model, $n$% of the ground truth data should lie in this interval. When plotting the different percentiles against the proportion of data points falling in the corresponding interval, well-calibrated predictions lie on the diagonal. Here, we evaluate this for the 60th, 80th, 90th, and 95th percentiles, which roughly correspond to one to three standard deviations. However, other evaluations, including percentile bins from 0 to 100, are also common (see e.g., Wei et al.[50]). If a model is overconfident, the corresponding values fall in the lower triangle below the diagonal. In the underconfident regime, i.e., if the average predictions are accurate, but the estimated uncertainty is too high, the values fall in the region above the diagonal. In all three datasets, we evaluate the calibration of masked continuous variables (GLVM, fly walking behavior, and behavior in the monkey reach task). The test set sizes, as well as the computational cost of estimating confidence intervals, differ between models. Therefore, we chose different $n_z$ and $n_{obs}$ to compute the confidence intervals but kept both values the same when evaluating the naive model to ensure a fair comparison.

**For count data**, we compute cumulative distribution functions (CDFs) of the spike counts to assess the calibration of the predicted firing rate. This is because most bin counts are either 0 or 1, making it impractical to construct confidence intervals.[55] More specifically, to obtain informative CDFs, we aggregate five neighboring bin counts of time-series with 200 time bins. Then, for each of the $n_z \cdot n_{obs}$ predicted rate time series, we sample spikes and compute the CDF over all 40 aggregated bins. Finally, we plot the obtained CDFs against the analogously aggregated CDF of the ground truth spike train. If the rate predictions are well calibrated, their resulting CDFs closely match the ground truth CDF and lie on the diagonal.

**Decoding performance vs. latent uncertainty** We evaluate the decoding performance of VAE latents obtained from partial neural recordings together with the VAE latent uncertainty in the most informative latents. For each of seven masking patterns (all observed, 5, 20, 50, 100, 150, or 200 masked neurons out of 213), we plot the mean latent standard deviation of the most informative latent against the corresponding decoding performance, fitting the relationship using linear regression. We can then evaluate the relationship for each model using the $R^2$ values, slopes, and associated p-values.