IDENTIFYING BEHAVIORAL STRUCTURE FROM DEEP VARIATIONAL EMBEDDINGS OF ANIMAL MOTION

Kevin Luxem[1,2], **Falko Fuhrmann**[2], **Johannes Kürsch**[1,2], **Stefan Remy**[1,2*] **and Pavol Bauer**[1,2*]
[1]Leibniz Institute for Neurobiology, Department of Cellular Neuroscience, Magdeburg, Germany
[2]German Center for Neurodegenerative Diseases, Neuronal Networks Group, Bonn, Germany
[1]firstname.secondname@lin-magdeburg.de
[2]firstname.secondname@dzne.de

October 28, 2020

ABSTRACT

Naturalistic behavior is highly complex and dynamic. Approaches aiming at understanding how neuronal ensembles generate behavior require robust behavioral quantification in order to correlate the neural activity patterns with behavioral motifs. Here, we present Variational Animal Motion Embedding (VAME), a probabilistic machine learning framework for discovery of the latent structure of animal behavior given an input time series obtained from markerless pose estimation tools.

To demonstrate our framework we perform unsupervised behavior phenotyping of APP/PS1 mice, an animal model of Alzheimer disease. Using markerless pose estimates from open-field exploration as input VAME uncovers the distribution of detailed and clearly segmented behavioral motifs. Moreover, we show that the recovered distribution of phenotype-specific motifs can be used to reliably distinguish between APP/PS1 and wildtype mice, while human experts fail to classify the phenotype based on the same video observations. We propose VAME as a versatile and robust tool for unsupervised quantification of behavior across organisms and experimental settings

***K*eywords** Neuroscience · Behavior Quantification · Machine Learning · Variational Bayes · Manifold

1 Introduction

Behavior is defined as the way in which an animal responses to a particular situation or stimulus, shaped by experience and knowledge (Carew, 2005). As of today, most studies investigating behavioral changes in model organisms rely on ethological classification performed by humans, which are mostly based on standardized protocols (Crawley, 2008). While standardization is important and ensures generalizability, the variability introduced by human annotators remains a potential confounding factor in the interpretation of behavioral phenotyping results from different laboratories (McIlwain, Merriweather, Yuva-Paylor, & Paylor, 2001). Moreover, in most currently used tests the behavioral repertoire of animals is reduced to easily quantifiable behavioral choices by task extensive pre-test training.

Thus, the need for robust unsupervised behavioral quantification methods has been widely recognized and innovative approaches in this direction are currently introduced (Gomez-Marin, Paton, Kampff, Costa, & Mainen, 2014; Brown & de Bivort, 2018). Unsupervised behavior quantification may not only provide a more unbiased description of naturalistic behavior, it may also be sensitive enough to detect subtle differences that would potentially remain undectedable or unquantifiable by a human experimenter (Anderson & Perona, 2014; Datta, Anderson, Branson, Perona, & Leifer, 2019). Furthermore, behavioral quantification based on high temporal resolution time series data may permit the computing-intensive analysis of correlations between behavior and neuronal activity. It may thus serve as an important tool that facilitates the discovery of causal relationships between brain activity and behavior (Markowitz et al., 2018; Musall, Kaufman, Juavinett, Gluf, & Churchland, 2019).

*Equal contribution of last and second last author.

Several computational approaches for unsupervised behavior quantification have been introduced (Berman, Choi, Bialek, & Shaevitz, 2014; Wiltschko et al., 2015; Batty et al., 2019). These methods advanced the field of unsupervised behavior quantification and established an increasing awareness of the necessity to improve objectivity. Most approaches operate on a dimensionality-reduced signal extracted directly from the tracking video. The signal is then learned by a machine learning model in the time-domain (Wiltschko et al., 2015; Batty et al., 2019) or frequency-time domain (Berman, 2018) and segmented into discrete blocks containing similar chunks of input data.

Recently, pose estimation tools such as *DeepLabCut* (Mathis et al., 2018) and *LEAP* (T. D. Pereira et al., 2019) enabled efficient tracking of animal body-parts via supervised deep learning. The robustness of deep neural networks allows the application of these tools for pose estimation in many model systems, such as mice, zebrafish and flies, and allows for a high generalization betweeen datasets (Mathis et al., 2018). However, while such tools provide a continuous representation of the animal body motion, the extraction of underlying discrete states as a basis for classification (Tinbergen, 1951) remains a challenge.

Here we introduce Variational Animal Motion Embedding (VAME), a probabilistic machine learning framework for clustering of behavioral signals obtained from pose estimation tools in both the spatial and the temporal domain. We propose that these continuous spatiotemporal signals can be grouped into discrete states via clustering of the latent vector obtained from a recurrent neural network autoencoder. Our approach is inspired by recent advances in the field of temporal action segmentation (Kuehne, Richard, & Gall, 2020), representation learning (Chung et al., 2015; Chen et al., 2016; Higgins et al., 2017; Jiang, Zheng, Tan, Tang, & Zhou, 2017) and unsupervised learning of multivariate time series (J. Pereira & Silveira, 2019; Ma, Zheng, Li, & Cottrell, 2019).

Our machine learning model is built within the framework of variational autoencoders (VAE) (Kingma & Welling, 2014; Rezende, Mohamed, & Wierstra, 2014) that has been previously applied to the field of neuroscience (Speiser et al., 2017; Pandarinath et al., 2018). Based on this work, it has been suggested that learning a disentangled representation of the data generative factors can be helpful for a large variety of tasks (Bengio, Courville, & Vincent, 2013). Using the recurrent autoencoder in a variational setting allows a model to learn a complex distribution of the data and to generalize well to previously unseen data. Moreover, VAEs enable the generation of synthetic samples from the learned distribution, that can be used to validate the learning process.

In this manuscript we first demonstrate the model and how it can be applied to behavioral data obtained during open-field exploration. We then demonstrate the sensitivity of our approach in a use case, in which we apply VAME to investigate behavioral differences in a mouse model of beta-amyloidosis (APP/PS1dE9) (Jankowsky et al., 2004). This mouse model and comparable humanized models are commonly used for preclinical studies. Detecting body pose signatures as functional "biomarkers" of underlying pathophysiology would enable early non-invasive detection and potentially facilitate preclinical research on early therapeutic intervention. We show that our method robustly identifies differences in distribution of behavioral motifs in transgenic versus wildtype animals. Furthermore, we used the learned representation to predict the phenotype of inidividual mice based exclusively on their behavioral motif distribution.We show that this classification performance is better than the classification based by human experts that have been presented the video data. Finally, we validate our model with human annotator generated labels and inspect the variability in clustering depending on the model parameters. Furthermore we evaluate and quantify the added value of adding temporal information in comparison to relying exclusively on spatial representations of the egocentric body pose of mice.

2 Results

2.1 VAME: Variational Animal Motion Embedding

There is a broad agreement in recent work on computational behavioral quantification that observable behavior can be encoded in a low-dimensional subspace or manifold (Wiltschko et al., 2015; Brown & de Bivort, 2018; Berman, 2018). Within this latent structure the identification of different behavioral motifs ranging from stereotyped behavior to rare or spontaneous events is a realistic goal.

To investigate behavioral structure we let animals move freely inside an open-field arena (Figure 1 B, top). During the experiment the animal movement was recorded from a camera that was mounted below the arena. Another behavior setup that can be straight-forwardly equipped with tracking cameras is the restrained setup where head-fixed animals behave on a linear treadmill. The quantification of such data using a related quantification method has been discussed in a conference proceeding previously (Luxem, Fuhrmann, Remy, & Bauer, 2019).

In order to identify the postural dynamics of the animal from the video recordings we used a markerless pose estimation tool (Mathis et al., 2018). From pose estimation we obtained a time-dependent series of marker positions $\mathbf{X}$ which captures the movement of relevant body parts. Our goal was to extract useful information from the time series data,

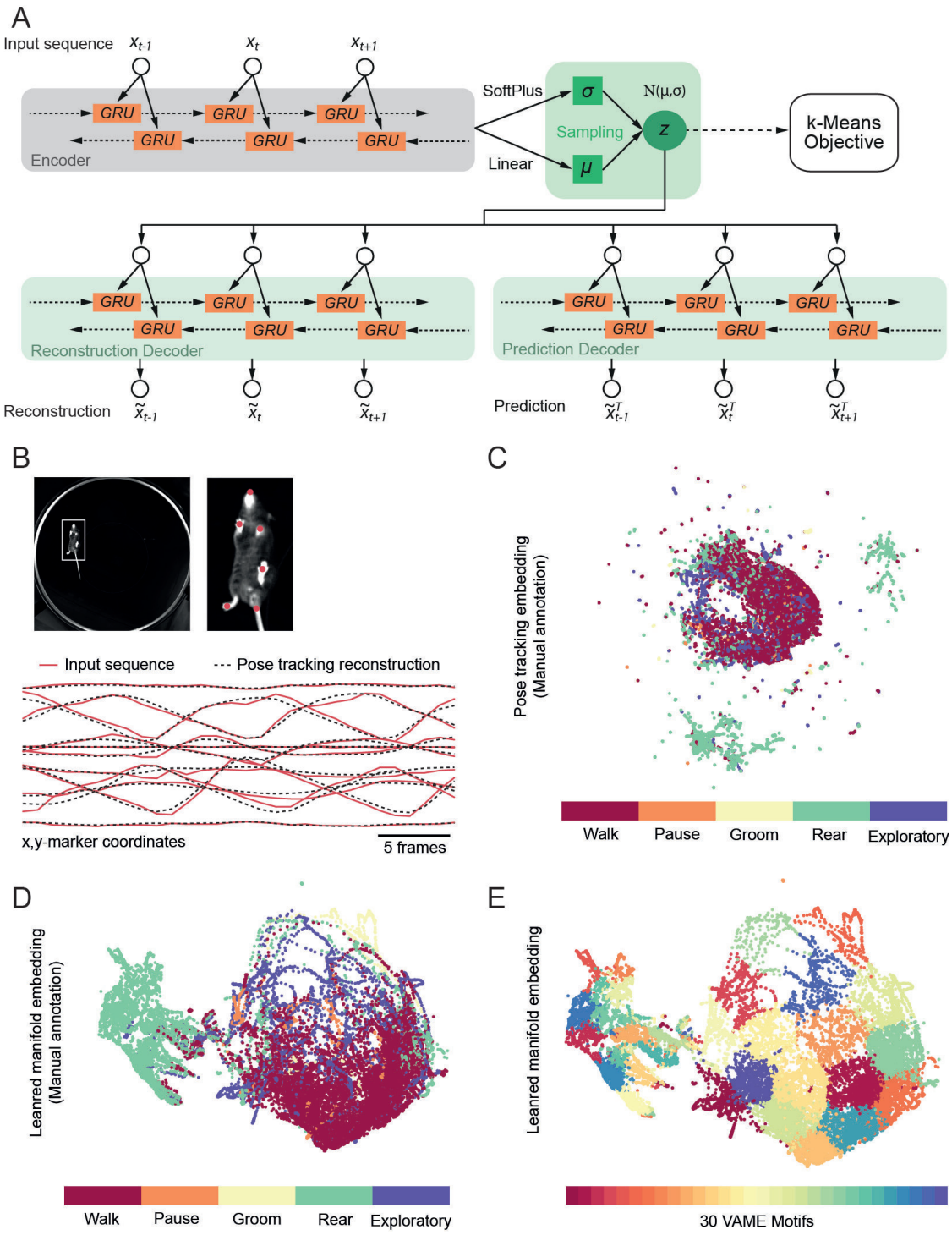

Figure 1: **VAME: a machine learning model for unsupervised spatiotemporal behavior quantification.** (A) Architecture of the machine learning model. The input sequence is fed into a bidirectional recurrent neural network (encoder) which learns a probabilistic representation **z** of the underlying hidden states. Another recurrent neural network (decoder) learns to reconstruct the original sequence from **z**. An additional prediction decoder can be used to anticipate the evolution of the time series. (B) Upper: Markerless pose estimation of animals performing in open-field setup, where marker positions are aligned to egocentric coordinates. Lower: Exemplary input time series reconstructed from the recurrent neuronal network decoder. (C-E) Unifold Manifold Approximation (UMAP) embeddings of the marker position input series (C) and the latent representation encoded from the RNN color-coded for 5 manually labeled behavioral classes (D). (E) Same as (D) but color-coded according to assignment into 30 VAME motifs.

that allowed for an effective behavioral quantification given spatial and temporal information of body movement. We aligned the marker positions egocentrically to the mouse body. Then, the aligned data was used as input to our machine learning model (Figure 1 B).

Within the framework of variational autoencoders (VAEs) (Kingma & Welling, 2014) we built a bidirectional recurrent neural network (RNN) encoder which was trained in an unsupervised fashion (Figure 1 A). Gated recurrent units (GRUs) (Cho et al., 2014) were used as the basic building block of the RNNs. The encoder receives a sample $\mathbf{x}_i$ of the time series and learns to embed the relevant information into a lower dimensional representation $\mathbf{z}_i$. Learning is achieved by passing $\mathbf{z}_i$ to another RNN which decodes the lower dimensional vector into an approximation $\tilde{\mathbf{x}}_i$ of the input chunk (Figure 1 B, bottom). Additionally, a second RNN decoder learns to anticipate the structure of the subsequent time series chunk $\tilde{\mathbf{x}}_{i+1}$ from $\mathbf{z}_i$ (Srivastava, Mansimov, & Salakhudinov, 2015), thereby regularizing $\mathbf{z}_i$ and forcing the encoder to learn a richer representation of the behavior. The prior of the VAE followed the standard normal distribution. However, inspired by Ma and colleagues (Ma et al., 2019), we introduced an additional prior on $\mathbf{z}_i$ with the aim to improve the clusterability of the latent space (see Methods 4.3 for details).

In order to investigate if our model learned a meaningful representation of the input time series, we visualized $\mathbf{z}$ using Unifold Manifold Approximation (UMAP) (Figure 1 D, E). Compared to the UMAP embedding of the egocentrically aligned spatial time series (Figure 1 C) the visualization of $\mathbf{z}$ suggested that the information from $\mathbf{x}$ is mapped into a dense manifold representing the spatiotemporal dynamics of the animal's behavior. We then assigned points of $\mathbf{z}$ to clusters while minimizing their within-cluster variance (k-Means algorithm). In this way we grouped input chunks by spatiotemporal similarity, i.e. created behavioral motifs (Figure S.1).

As the VAE is a generative model, it is furthermore possible to generate unseen data from the latent space learned by the model. We demonstrate this capability in Figure S.5, where we perform *latent interpolation* between two time windows randomly chosen from the input data. Given the start and end point, the model is capable to interpolate between the data by generating the most likely output time series based on the learned data representation.

2.2 Unsupervised detection of behavioral differences between phenotypes

To demonstrate how VAME can be applied for the detection of individual animal specific as well as cohort specific behavioral differences we performed behavioral tracking on a mouse model of beta-amyloidosis harboring human mutations in the APP and presenelin 1 gene (Jankowsky et al., 2004). Mice heterozygous for the transgenic allele were compared to wildtype littermate control. For this mouse line, several age-dependent behavioral differences have been reported (Huang et al., 2016). For example, age/disease related motor and coordination deficits (Onos et al., 2019), changes in anxiety levels (Lalonde, Kim, & Fukuchi, 2004) and spatial reference memory deficits (Janus, Flores, Xu, & Borchelt, 2015) were observed. This dataset forms an ideal use-case for the purpose of unsupervised behavior quantification as both phenotypes show considerably similar observable behavior, although the execution of specific motifs is expected to be altered by the underlying pathomechanism.

We placed N=8 mice into a novel open-field environment with transparent bottom, in which mice were allowed to freely explore the arena for a duration of 25 minutes after an initial habituation period of 10 minutes (Figure 2 A). During the experiment, the nose, tailroot, hind and front paw movement was captured by a camera mounted below the arena (Figure 1 B, top).

The average speed during the trial was 2.29 ± 1.57 cm/s for control animals and 2.49 ± 2.45 cm/s for test animals, while the average distance travelled was 9187.44 ± 1266.4 cm and 9937.07 ± 1367.08 cm, respectively (Figure 2 C). No statistically significant differences were computed between the groups for both measures. Observing the total occupancy for control and test animals, we found that both groups preferred the boundary of the arena over the middle, while the test animals had an additional bias towards the southern border over the northern (Figure 2 B).

We extracted marker coordinates from pose tracking, trained the machine learning model on the full dataset (1.3×10^6 time points) and clustered the multivariate signal into 30 VAME motifs. From the transition probabilities between individual motifs we created a hierarchical representation of mouse behavior (Figure 2 D). We iteratively grouped two motifs that had the largest transition probability between each other as well as the smallest joint probability of occurrence (see Methods 4.4 for details). Doing so, we made sure that the motifs having the largest spatiotemporal similarity have been grouped at the lowest levels of the hierarchy. We then cut the tree-like hierarchical structure on the second and third hierarchical level in order to obtain communities of motifs. Figure 2 D shows the obtained representation for a single wildtype animal. Note that the structure of the obtained representation was similar for all animals.

Moreover, the hierarchical representation offers a practical solution for the problem of overclustering. As the number of cluster k is not known, it is likely that we choose k to be larger as in the real data distribution, thus effectively

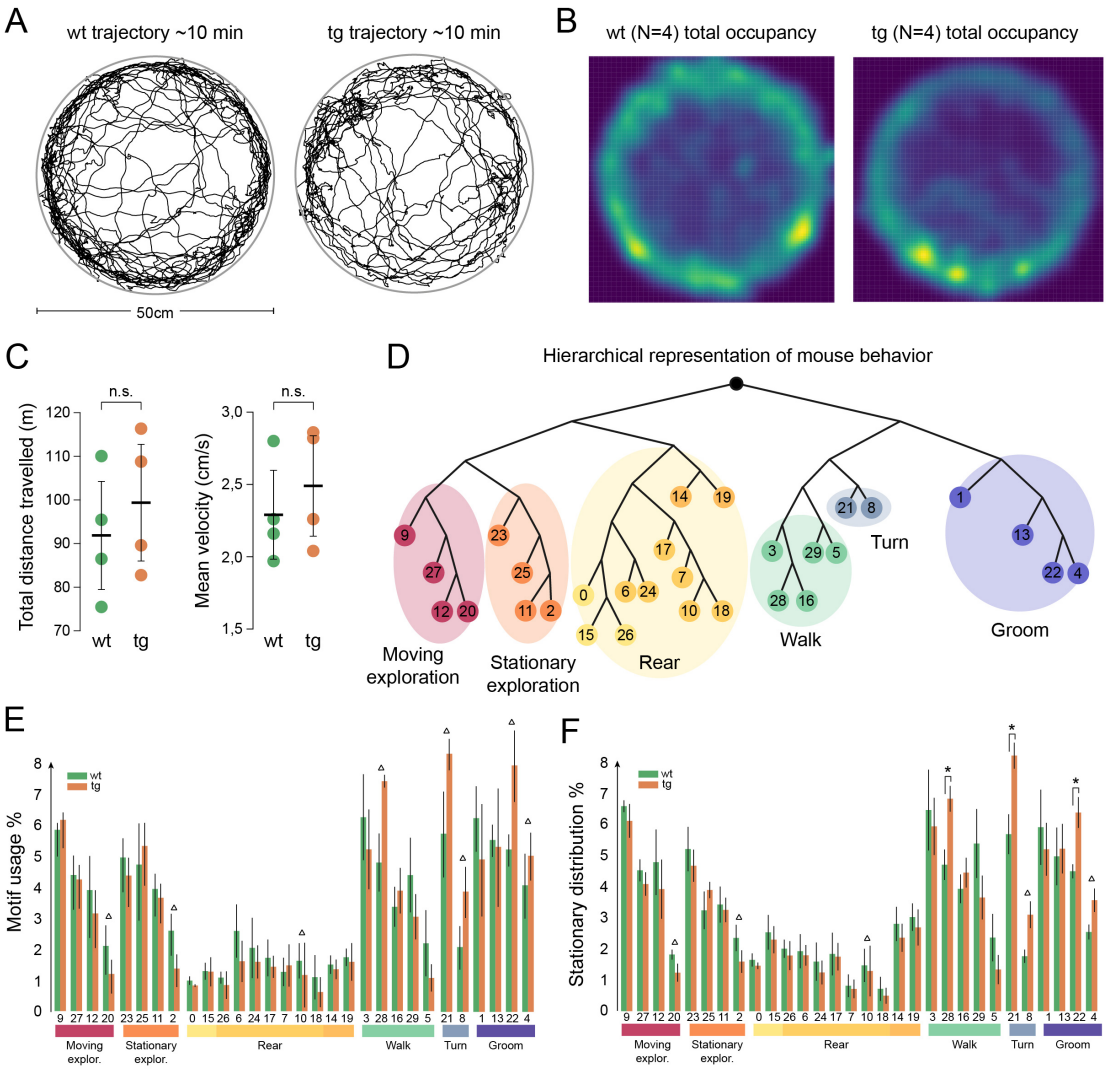

Figure 2: **Unsupervised detection of behavioral differences between two phenotypes.** (A) Wildtype (wt, N=4) and APP-PS1 mice (tg, N=4) performing an open-field exploration task for 25 minutes. (B) Spatial occupancy for the whole duration of the experiment. (C) Average speed and distance travelled. (D) Hierarchical representation of VAME motifs obtained from $k = 30$ clusters with post-hoc annotation. (E) Percentage of usage of each VAME motif for the control and test group. Motifs marked with Δ are discussed in text and visualized in Figure 3. (F) Stationary distribution probability obtained from the Markovian transitions between motifs for the control and test group. Significant differences were found in 3 motifs (Asterisks, p (Bonferroni-adjusted) < 0.05).

partitioning the space into roughly equally large chunks. However, we can recover the macroscopic structure of behavior from the hierarhical representation, as suggested in the UMAP visualization in Figure 3 A. After clustering the data of one animal into 30 VAME motifs and obtaining their respective communities we observe that the latent space contains now more self-contained patches of the communities.

We post-hoc labeled the communities into coarse labels that have been used for manual annotation of behavior before, with the addition of a *Turn* community that emerged in the hierarchical representation. In both motifs included in the *Turn* community we could detect bending and turning behavior of the animal that occurred in a stationary position or slow movement speed. Moreover, we have identified two communities containing motifs related to exploratory behavior. In both communities, the animal exhibited head movement strongly resembling undirected sniffing. However, in one community, the animal was mostly in an otherwise stationary position while in the other the animal was moving, typically at a slow pace. Note that the *Groom* community also contained motifs exhibiting consummatory behavior, as we have randomly placed three chocolate flakes in the center of the arena prior to the experiment in order to motivate full coverage of the open field.

Inspecting the difference in motif usage between control and test animals, we detected 3 VAME motifs that significantly within the stationary distribution differed between the groups. Of those, one was post-hoc categorized into the *Groom* community (motif 22, p (Bonferroni-adjusted) = 0.03, two-sample T-test with Bonferroni-correction), one in the *Turn* community (motif 21, p (Bonferroni-adjusted) = 0.04) and one as walk behavior (motif 28, p (Bonferroni-adjusted) = 0.04) (Figure 2 F). Grooming motifs 4 and 22 are characterized by no movement of hind paws, upper body and head movements, and slightly lifted front paws, as well as consumatory behavior (Figure 3 A, Supplementary Video 1). Sequences within motifs 8 and 21 display slow body transitions with hind paw movement, upper body and head turns, as well as low rears (Figure 3 A, Supplementary Video 2). Motif 28 shows moderate walking with the snout approaching the floor (Figure 3 A, Supplementary Video 3). The dependency of the discovered number of significantly changed motifs within the stationary distribution on the cluster size k is further visualized in Figure S.6.

Furthermore, motifs within the *Moving exploration* community show walking with a slightly bend upper body and active sniff behavior with head movement (Figure 3, Supplementary Video 4) while in motifs within the *Stationary exploration* community mice perform intense sniff behavior while sitting on the hind paws (Figure 3 A, Supplementary Video 5). Interestingly, although not statistically significant, the pronounced appearance of exploratory motifs together with the significantly more frequent motif from the *Turn* commuity could be related to deficits of spatial orientation that have been previously reported for comparable mouse models (Lalonde et al., 2004; Janus et al., 2015).

To further visualize motion patterns of individual motifs we have carried out a Fourier analysis of virtual marker movements (Figure 3 B). In this visualization, we show how movements occur in different frequency ranges between 1 and 10 Hz. Here we observe that higher frequencies typically dominate the power spectrum for hind paw movements within the *Walk* community, while motifs in the *Rear* community typically show activity of 5 Hz or higher for the front paws. Moreover, in Figure 3 C, we see how motifs develop over time, and that the average usage in 2.5 minute long bins is approximately stable over the duration of the experiment.

2.3 Phenotype classification

Finally, we investigated if the wildtype and APP/PS1 transgenic mice could be distinguished based on the motif usage and stationary distribution obtained from VAME. When measuring the Kullblack-Leibler divergence between the motif usage distributions as well as the stationary distribution per animal we found a block-diagonal structure in the pairwise dissimilarity matrix (Figure 4 A, B). This suggests that two separable clusters exist. Interestingly, when applying k-Means clustering with the cluster size $k = 2$ to the underlying distributions indeed a decision boundary could be found that seperated each animal into a cluster corresponding to the correct genotype. The robustness of the classification was underpinned by Leave-p-out validation. When leaving $p = 1$ datasets out of the clustering procedure the mean accuracy was 0.96 ± 0.06, while it has been found to be 0.91 ± 0.12 and 0.88 ± 0.14 for $p = 2$ and $p = 3$, respectively.This finding could be furthermore confirmed with statistical testing, when KL-distances within a genotype were compared to between-genotype KL-distances (Figure 4 C: Ks-test $p = 0,02$, t-test wt to tg $p = 0.52$, t-test wt to between group KL-distance $p = 0,0002$, t-test tg to between group KL-distance $p = 0.000002$, Figure 4 D: Ks-test $p = 0,002$, t-test wt to tg $p = 0.49$, t-test wt to between group KL-distance $p = 0,0004$, t-test tg to between group KL-distance $p = 0.00002$). Note that the Kullblack-Leibler divergence is a measure of distances between probability distributions. Thus, the within-group statistics appearing in Figure 4 C, D is smaller than the between-group statistics given the separation of phenotypes into distinct clusters is feasible in this case.

In order to investigate how the cluster size affects the separability of both phenotypes, we performed an ablation study where we computed the cluster distance for each group to their k-Means centroid. Figure 4 G, H shows the mean and standard deviation from this centroid for each group. We observe that the seperation of both phenotypes can be reliably

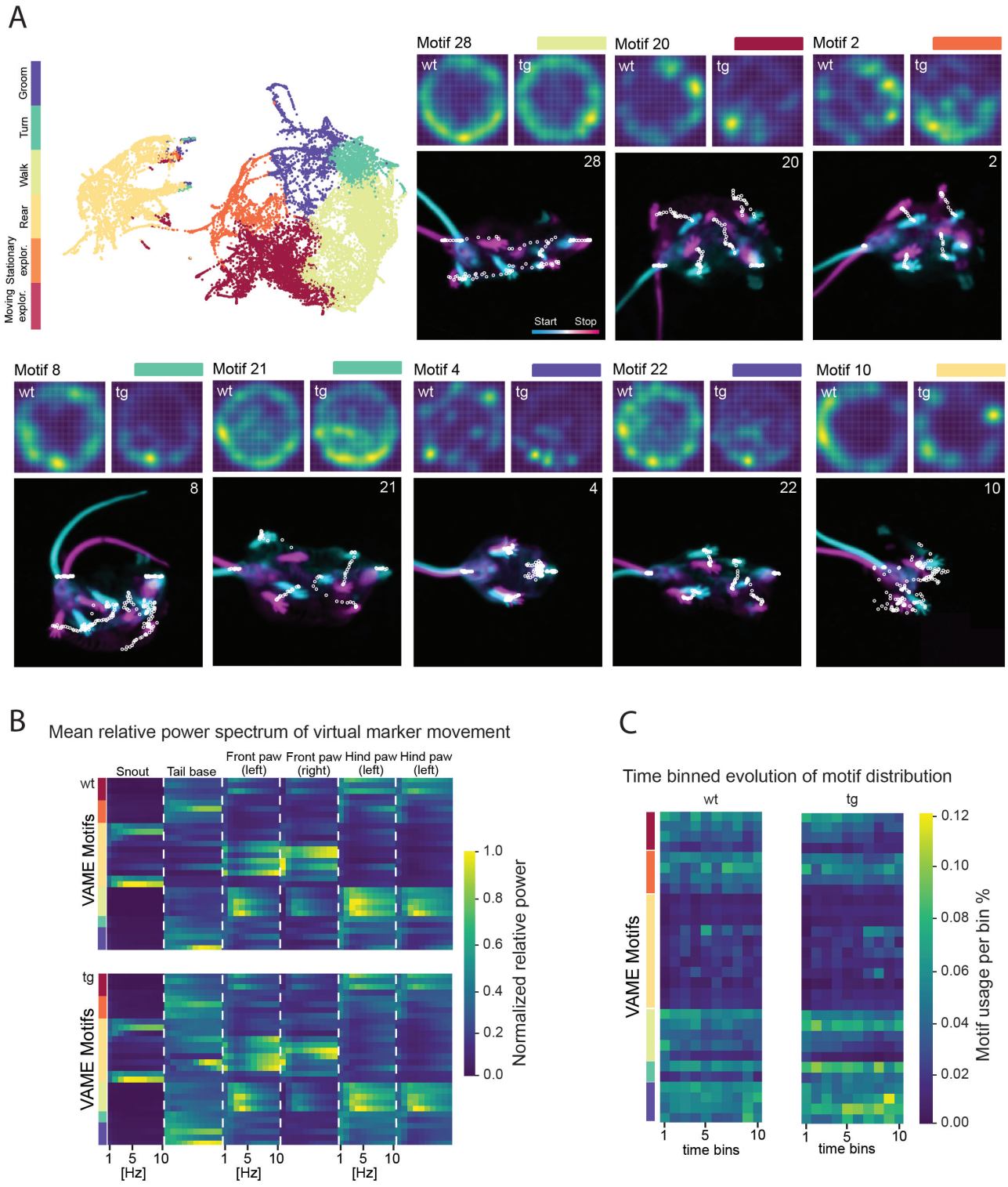

Figure 3: **Representation of VAME motifs detected during the open-field exploration task.** (A) Top Left: Hierarchical representation assignment visualized on the UMAP embedding. Others: Exemplary VAME motifs detected during the open-field exploration task. Motif images are ordered as follow - Upper: Density of the motif occurrence for wildtype and transgenic mice. Lower: Animal body pose at the beginning of the motif (blue) and at the end of the motif (red). White dots represent the movement of marker positions obtained from pose estimation during the motif. (B) Mean power spectrum of virtual marker movement detected on different body parts during the occurrence of VAME motifs for both wildtype (Upper) and control (Lower) animals. (C) Mean motif usage for bins of 2.5 minute duration for both phenotypes.

done with more than 10 cluster. We moreover inspected how the variance of cluster distances changes given different recording times. More specifically, we have segmented our recordings into 5, 10, 15 and 20 minutes (see Figure S.3) and carried out the same analysis as in Figure 4 G, H. In this ablation study we found that a minimum recording time of 15 minutes is necessary to reliably separate both groups.

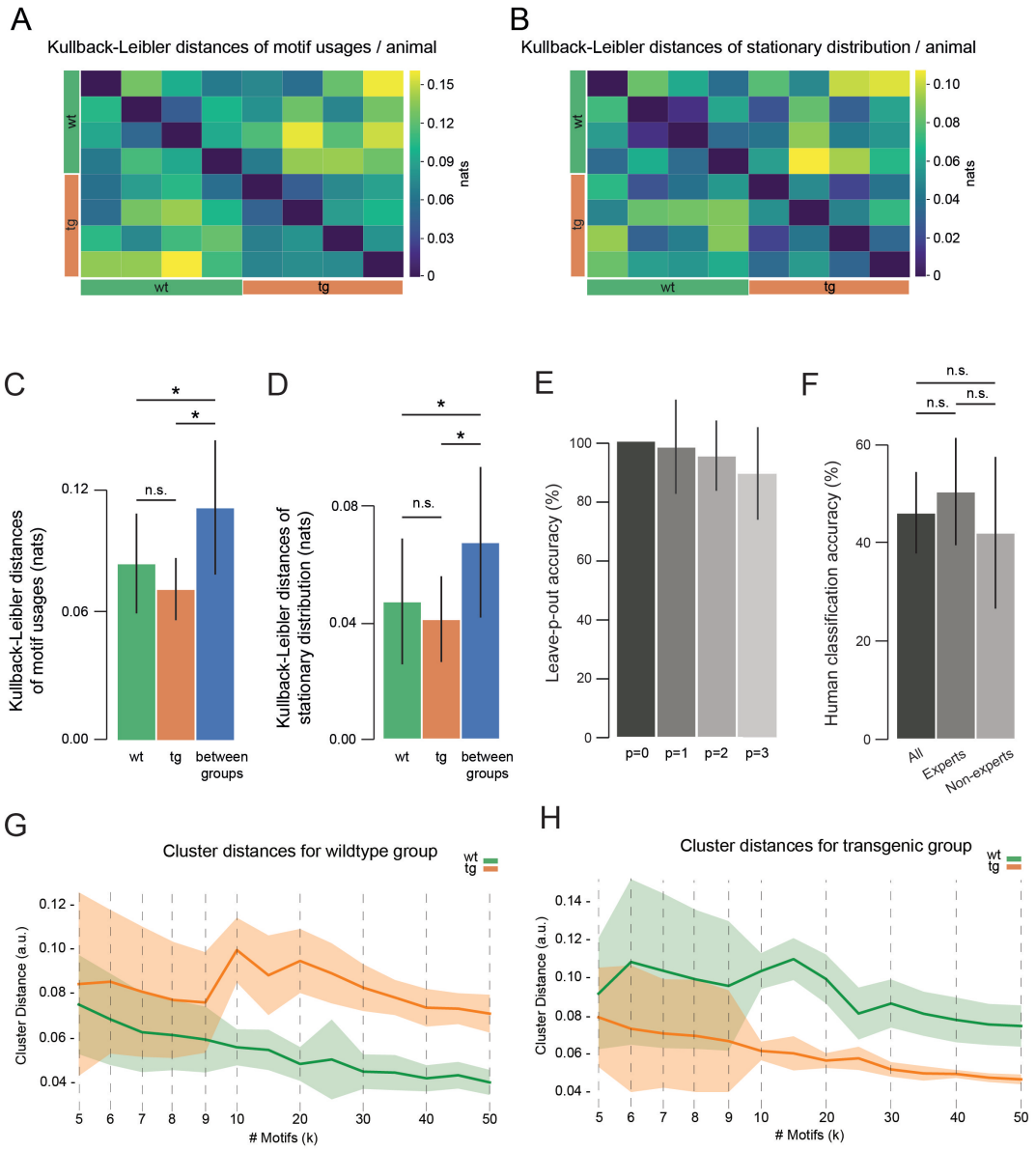

Figure 4: **Classification of phenotypes using VAME** (A,B) Kullback-Leibler distance of motif usages (A) and the stationary distribution (B) between animals. (C,D) Group statistics of Kullback-Leibler distances of motif usage (C) and the stationary distribution (D) within and between phenotypes. (E) Leave-p-out accuracy for unsupervised classification given p held-out datasets (Total N=8). (F) Accuracy of the human classification task (Asterisks, $p < 0.01$). (G,H) Distances to between-phenotype cluster centroids given each datapoint consists of k motifs, for the wildtype (G) and transgenic group (H). Shaded area corresponds to the standard deviation of distances from the k-Means centroid.

In order to demonstrate the strength of our approach in separation of behavioral phenotypes into genotypes we asked 11 human experts to classify the behavior based on the video recordings that were used as an input to the machine learning model. We constructed an online questionnaire for blinded classification where each participant was allowed to watch all videos for an unlimited time before making her decision. All experts had previous experience with behavioral video recordings in an open field and/or treadmill setting. In addition, six of the participants also had previous experience

doing behavioral experiments with APP/PS1 mice. We found that the later group showed slightly higher classification accuracy than experts inexperienced with the mouse model ($50.98\% \pm 11.04\%$ for experts, $42.5\% \pm 15.61\%$ for non-experts). However the overall human classification accuracy was at chance level for all participants ($46.61\% \pm 8.41\%$, Figure 4 E). Note, however, that the comparison to human performance is not fair, as humans made their judgment based on observations of the raw video material and did not have the possibility to assess the distribution of motifs, provided by, for example, an ethogram. Thus, we believe that community-based labeling of the complete video material would be a relevant follow-up to this comparison. Nonetheless, we believe that these result demonstrate the usefulness of VAME for constructing a motif distribution that captured even the subtle differences of behavior specific to the phenotype, that were not detectable by human experts given the same input data.

2.4 Validation of VAME

In order to validate how motifs obtained by the proposed unsupervised approach coincide with ethograms created by humans, we created a manually labeled dataset that was annotated by three experts with training in behavioral neuroscience. The experts annotated a video of a freely moving wildtype animal consisting of 20.000 frames (≈ 6 minutes length) with 5 coarse behavioral labels (Walk, Pause, Groom, Rear, Exploratory behavior) (Figure 5 A, see Methods 4.5). When quantifying agreement between individual experts, we observed that 71.93% of the video frames were labeled equally by all three experts. The remaining 13.61% of frames were labeled unequally by two experts and 14.47% were labeled unequally by all three experts (Figure 5 C). This implicated that behavior showed a considerably high observer variability and is not trivially assignable to discrete labels (Anderson & Perona, 2014; Datta et al., 2019).

Next, we obtained VAME motifs for the tracking data used in Figure 5 A,B and validated how they overlapped with the manual annotation (Figure 5 B, bottom). We found that although most VAME motifs predominantly overlapped with a single manually assigned label, a portion of the VAME motifs overlapped with two or more manually assigned labels. Interestingly, we found more disagreement between human experts for motifs that overlapped with several human assigned labels, indicating uncertainty of the annotators. This suggests that the clustering accuracy of VAME is high, but the achieved score is low as the underlying behavior can not be uniquely identified by experts.

We further used two different metrics for clustering validation to quantify the model accuracy compared to the manual annotation. Note that the goal of this benchmark is not to create a one-to-one mapping of VAME motifs to manually assigned labels akin to a supervised approach but to create a one-to-many assignment that may serve as validation of the proposed method as well as for the purpose of model comparison. Purity was used as a measure of the extent to which clusters contain a single manually assigned label (Manning, Raghavan, & Schütze, 2008). Normalized Mutual Information (NMI) was introduced as information-theoretic metric, that scales the amount of mutual information between VAME clusters and the manually assigned labels (see Methods 4.5 for details).

Compared to the clustering of the egocentrically aligned spatial input signal using the k-Means algorithm we found a relative increase of the Purity score for our best model (Spatio-temporal + Prediction) by $7.03\%, 8.56\%, 8.61\%$ relative to our model (absolute values: $75.11, 77.2, 77.77$) for each choice of the numbers of cluster $k = \{15, 30, 45\}$, respectively. Likewise we found a relative increase of the NMI score by $40.14\%, 47.23\%, 43.23\%$ (absolute values: $27.44, 27.93, 27$) for each choice of k, obtained by our best model (Spatio-temporal + Prediction). Absolute values for each setting and metric employed in the comparison are found in Table S.1. Furthermore, we have compared the scores obtained for VAME with scores obtained for clustering of singular values of the spatiotemporal signal (Table S.1). In comparison, we found that for all settings of k the Purity and NMI score is lesser then the scores obtained with our model. This finding is also confirmed for bootstrapped statistics of the Purity and NMI score where the measures have been computed for 1-minute long bins of the human labeled data and the corresponding VAME motifs, as shown in Figure S.2.

We moreover compared our approach to an autoregressive Hidden-Markov model (AR-HMM) which was introduced for the purpose of behavior quantification by Wiltschko and colleagues (Wiltschko et al., 2015). As shown in Table S.1, the AR-HMM model obtained a relative increase of the Purity measure by $4.18\%, 3.49\%, 3\%$ (absolute values: $73.07, 73.58, 73.75$) and a relative increase of the NMI measure by $16.8\%, 23.56\%, 25.62\%$ (absolute values: $22.87, 23.44, 23.68$) for each choice $k = \{15, 30, 45\}$, respectively. We also compared our approach to the Motion-Mapper framework proposed by Berman and colleagues(Berman et al., 2014). The framework suggested a clustering using a cluster size of $k = 17$ for the given dataset. As shown in Table S.2, the model obtained a relative increase of Purity by $4, 18\%$ (absolute value: 72.96) and relative increase of NMI by 28.29% (absolute value: 25.78).

Finally, we visualized the trajectories of three randomly chosen behavioral sequences with a length of 1.5 seconds within a lower-dimensional space that was projected by UMAP based either on the spatial input signal (Figure 5 F) or the spatiotemporal representation learned by our model (Figure 5 G). We observed that the course of trajectories within the embedding of the spatiotemporal representation followed a smooth path through the projected manifold while the

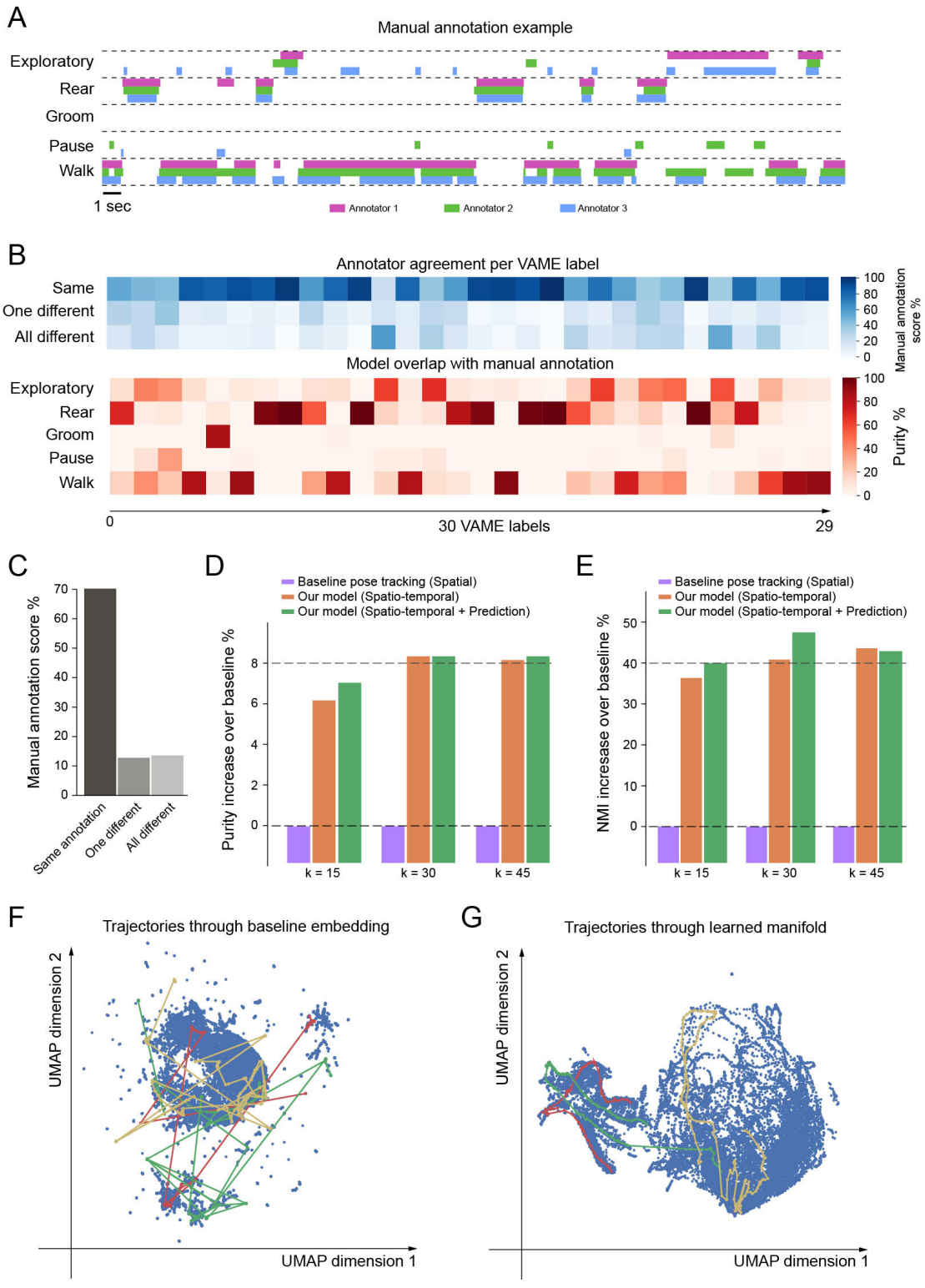

Figure 5: **Model validation via manually assigned labels on an open field dataset.** (A) Overlap of manually assigned labels by three experts. (B) Lower: Confusion matrix showing the agreement between 30 VAME motifs and 5 manually assigned labels. Upper: Agreement in manual annotation of behavior shown for the duration of each VAME motifs. (C) Disagreement in manual annotation. (D) Purity score obtained for three different choices of the number of clusters k for the spatiotemporal representation obtained with or without the prediction decoder relative to naïve clustering of the spatial input signal (baseline). (E) Same as (D) but showing the Normalized Mutual Information score increase compared to the baseline. (F, G) Three exemplary paths of consecutive video frames through the UMAP embedding space of the spatial input series (F) and through the embedding space visualizing the spatiotemporal representation (G).

course of trajectories through the embedding of the spatial input signal consisted of several scattered jumps through the projected space. Although a similar effect can be achieved considering, for example, points obtained from a wavelet transform, this suggests that our machine learning model captures the spatiotemporal dependencies of the input data and thereby unravels the development of observable behavior on a low-dimensional manifold.

3 Discussion

Animal movements occur in a broad range of spatial and temporal scales. The detection of stereotypical patterns is a classical ethological approach. However, within these patterns subpatterns exist which may continuously change in response to behavioral and environmental challenges and usually remain undetected. As manual scoring can not capture the full complexity of the behavioral dynamics that is required for detecting causal relationship of neural activity and behavior there is a pressing need for unsupervised behavior quantification in neuroscience.

To address this issues, we presented a probabilistic machine learning framework for clustering of spatiotemporal motion dynamics embedded in lower dimensional space (Variational Animal Motion Embedding, VAME). VAME allows for detection of behavioral motifs in time series data obtained with recently established deep-learning based pose estimation tools. Moreover, our approach offers the opportunity for embedding of other complementary modalities acquired at a similar temporal scale, for example the body temperature, blood oxygen levels, or other physiological parameters. Since the framework can be useful for the wider community of behavioral neuroscientists we provide open-source access to all required code and documentation. Moreover, we designed and used a rodent observation setup that is easily transferable to other laboratories. We demonstrated that a single camera observation from below the animal provides sufficient information for behavior quantification.

VAME requires only the setting of a few key parameters which is advantageous compared to previously introduced unsupervised quantification methods based on segmentation via autoregressive Hidden Markov models (Wiltschko et al., 2015; Batty et al., 2019). One important problem of Hidden Markov Models (HMMs) when applied to behavioral data is the short switching times between modules, following an exponential distribution. To circumvent this problem "sticky" autoregressive parameters have been introduced (Fox, Sudderth, Jordan, & Willsky, 2011). This step however requires the introduction of multiple additional potentially confounding parameters that are not trivial to set by users not familiar with advanced machine learning techniques. Moreover, RNNs have more expressive power than HMMs as their activation functions enable to capture non-linearities of the input data.

Our model of observable behavioral dynamics is a deep-learning based model that is trained in a fully unsupervised fashion. With the capability of deep neuronal networks to extract higher order features they can identify complex patterns within and across raw data points, thereby considerably reducing human effort needed to parametrize the model (Salinas, Flunkert, Gasthaus, & Januschowski, 2019). In order to achieve a high performance level they typically require larger amounts of training data in order to learn accurate models, as fewer structural assumptions are made than in parametric models including state space models. However, today the availability of extended data sets is not a major limiting factor as with the advent of markerless pose estimation tools continuous long-term monitoring of behavior is developing into a state-of-the-art approach in behavioral neuroscience.

Moreover we have decided to build our model within the class on Variational Autoencoders (VAEs) (Kingma & Welling, 2014). VAEs are generative models that combine probabilistic modeling and deep learning into one framework that enables the learning of the latent variable distribution underlying the input data. Moreover, we have adapted the classical VAE model with the addition of a prediction decoder, that anticipates the development of the learned time series and thereby regularizes the learning problem (Srivastava et al., 2015). Furthermore, we introduced an additional prior on the latent space in form of a k-means objective (Ma et al., 2019). We found that although the prior did not significantly improve the Purity and NMI scores in reference to the manual labeling, it increased the quality of the obtained clusters that were validated via generated video sequences (Supplementary Videos 1–5).

We have chosen the VAE as our underlying framework due to the capability of the model to create mapping between the data distribution and a latent space whose distribution follows the prior distribution. This is an advantage in comparison to a regular autoencoder (Kingma & Welling, 2019). In comparison to standard autoencoders, VAEs learn the parameters of the probability distribution, instead of just learning the "compressed representation" of the input data. On one hand this property ensures that the latent space where input vectors are embedded is continuous, thus allowing for smooth interpolation and clustering. On the other hand, this property allows to sample the latent space to generate new input data samples (Figure S.5). This can be useful to validate if the model learned a meaningful data distribution and furthermore can be used to generate synthetic behavioral data that may serve as input to other computational models or simulations.

Our method was inspired by the approach proposed by Berman and colleagues (Berman et al., 2014), that has been previously applied for unsupervised behavior quantification from marker time series (Günel et al., 2019). In the original paper (Berman et al., 2014), the authors applied signal processing techniques to extract relevant features describing animal movement, such as the leg segments of a fruit fly. Comparably, our approach relies on the pre-selection of features that either captures the full range of observed animal behavior or that are of specific interest for a given study, e.g. pupil dynamics or facial muscle movements. The previously published approach (Berman et al., 2014) then transformed the behavioral time series into a spectrogram, embedded it into a two-dimensional space via t-distributed stochastic neighborhood embedding (t-SNE) and obtains discrete modules using watershed transform of the continuous density map. Differently, our approach now uses a variational recurrent autoencoder, which is an alternative technique for non-linear dimensionality reduction. Both approaches aim at finding a lower dimensional embedding of the input data, but optimize a different term in order to achieve this goal. While variational autoencoders learn the parameters of the probability distribution that underlies the input data, t-SNE applies different transformations in different regions of the data in order to find a low-dimensional map that roughly preserves the distances in the high-dimensional space. For t-SNE applications a set of hyperparameters has to be tuned beforehand, otherwise the algorithm likely produces low-dimensional distributions that may misrepresent the global geometry of the input data (Kobak & Berens, 2019). Due to this, t-SNE is usually preferred for visualization purposes while variational autoencoders are preferentially applied for learning deterministic and reversible mapping from data to the embedding space. Moreoever, the quadratic computational complexity of t-SNE (van der Maaten and Hinton 2008) may be precluding for the creation of joint embeddings from large datasets, while the complexity of recurrent neuronal networks is asymptotically linear in the length of the input (Goodfellow, Bengio, and Courville, 2016) although novel approaches have been shown to generate t-SNE embeddings at linear time complexity via sub-sampling of datapoints (Cande et al., 2018).

In our approach we avoided the transformation of the input signal into the time-frequency domain. This allowed us to treat the signal in its raw form instead of finding a convenient balance of time and frequency resolution, that is necessary for an effective Fourier decomposition. However, it is also possible to extend our framework by incorporating, for example, a multimodal time series consisting of signals representing the wavelet power at a given frequency band (Berman et al., 2014) or traces obtained from filtering for other specific features of the input signal. Note also that an orthogonal approach to the proposed framework is behavior quantification based on automatically extracted deep features, as proposed by Batty and colleagues (Batty et al., 2019). We believe that the application of VAME to learning of dependencies in deep features could indeed yield a more expanded discovery of behavioral motifs.

A relevant parameter in our model is the size of the temporal window that slides over the input data during training and prediction. Shorter settings of the time window lead to learning of finer nuances within the signal, while larger time windows are required to capture long term dependencies. In our analysis, we have used temporal windows of 500 ms as input to our model, a setting which was proposed based on changepoint analysis of mouse behavior in previous work (Wiltschko et al., 2015). Furthermore the prediction decoder predicts the evolution of the next 250 ms of the input signal, serving as a regularization term. Depending on the time scale of movements across model organism this parameter has to be specifically set to capture relevant features. Note, that the RNN is capable to learn significantly longer temporal dependencies than given by a 500 ms window and this advantage could be potentially further utilized by future approaches of behavior quantification. Our approach can be potentially further improved by using the recently proposed dilated RNNs (Chang et al., 2017) as an encoder model which could lead to an improved treatment of the signal on multiple temporal scales. However, as the approach may require precise tuning of hyperparameters, the comparison to VAEs was not a feasible strategy for this study.

Another relevant parameter requiring optimization is the size of the latent vector, which is used to set the amount of information compression between the encoder and decoder networks. A too large size of the latent vector leads to poor performance of the encoder as it can surpass the inference step and leads to storage of the complete input information directly in the latent vector. A too small size of the latent vector on the other hand may not offer sufficient capacity for encoding relevant information, even after extensive training of the model. This parameter was set empirically to $\mathbf{z} \in \mathbb{R}^{30}$ while comparing the difference between input and reconstructed signals. For the appropriate setting the high-frequency noise was removed while the reconstructed signal captured the main characteristics of the input signal (see Figure 1B, Lower). Clearly, this setting requires adjustment for specific use cases and needs to be evaluated when the dimensionality of the input signal changes. For the given data, an ablation study of the latent vector size with respect to the reconstruction error can be found in Figure S.4. This data suggests that the reconstruction error lies within the same order of magnitude for the range of 10 to 35 latent variables, but is generally lower when the prediction term is employed in the loss function.

Given that mouse behavior takes place in three dimensions it could be beneficial to enhance the tracking setup with additional cameras enabling 3D-keypoint detection via triangulation (Günel et al., 2019). The resulting three-dimensional time series can then be fed into the proposed machine learning model akin to the datasets presented herein. This could lead to a better resolution of behavioral motifs, allowing for more detailed detection of behavior

patterns during rearing or grooming episodes. We summarize the most important choices of recording methods and hyperparameter settings in Table S.3.

We have demonstrated how our approach can be applied in practice for robust identification of the detection of differences between two mice of different genotypes. Classification of behavioral phenotypes based on defined features has been demonstrated previously, as for example in *Caenorhabditis Elegans* (Baek, Cosman, Feng, Silver, & Schafer, 2002). We would like to point out, however, that our classification is based on unsupervised discovery of behavioral motifs, with the only manual selection of features being the keypoints defined for pose estimation. Our model thus outperformed human clustering performance, demonstrating the practicability of our approach and, in general, machine learning based behavior quantification approaches in neuroscience.

Finally, we have validated VAME based on manual labeling and addressed the issue of behavior identifiability, that has been raised previously (Anderson & Perona, 2014; Datta et al., 2019). Here, we labeled behavioral data based on a composition of stereotypical movements, such as walking, pausing and sniffing, that may also appear in combination with each other. However, we found considerable disagreement in labeling between individual human experts, even if the basic set of motifs was grouped into five coarse behavioral classes, for example a motif jointly representing walking and exploratory behavior. While undirected sniffing in a rigid body pose is typically interpreted as exploratory behavior, sniffing during walking may be identified as regular walking behavior. At this point our approach reached a limit. Possibly, this limit may only be overcome by an identification of the neuronal correlates of the true behavioral states (Krakauer, Ghazanfar, Gomez-Marin, MacIver, & Poeppel, 2017).

We are thus convinced that our framework will be useful to robustly detect behavior representations across organisms and experimental settings. Moreover, we anticipate that VAME will stimulate the development of further machine learning models that may be benchmarked provided the data and methodology presented in this paper. Finally, with the introduction of VAME we aim at facilitating the initiation of studies investigating causal relationships between naturalistic behavior and neuronal activity.

4 Methods

4.1 Animals

For all experiments we used 12 month old male transgenic and non-transgenic APPSwe/PS1dE9 (APP/PS1) mice (Jankowsky et al., 2001) on a C57BL/6J background (Jackson Laboratory). Mice were group housed under standard laboratory conditions with a 12-h light-dark cycle with food and water ad libitum. All experimental procedures were performed in accordance with institutional animal welfare guidelines and were approved by the state government of North Rhine-Westphalia, Germany.

4.2 Experimental setup, data acquisition and preprocessing

For the open field exploration experiment mice were placed in the center of an circular area (transparent Plexiglas floor with diameter of 50 cm surrounded by a transparent Plexiglas wall with height of 50 cm) and have been left to habituate for a duration of 10 minutes. Afterwards, sessions of 25 minutes were recorded where the mice were left to freely behave in the arena. To encourage a better coverage, three chocolate flakes were placed uniformly distributed in the central part of the arena prior to the experiment.

Mouse behavior was recorded by a CMOS camera (Basler acA2000-165umNIR) equipped with wide angle lens (CVO GM24514MCN, Stemmer Imaging) that was placed centrally 35 cm below the arena. Three infrared light sources (LIU780A, Thorlabs) were placed 70 cm away from the center, providing homogeneous illumination of the recording arena from below. All recordings were performed at dim room light conditions.

For behavioral pose extraction, m virtual markers were placed on relevant bodyparts in 650 uniformly picked video frames from 14 videos in the freely behaving setup. A residual neural network (ResNet-50) was trained to assign the virtual markers to every video frame (Mathis et al., 2018). The resulting training error was 2.14 pixels and the test error 2.51 pixels, respectively.

To obtain the egocentric time series of (x, y) marker coordinates aligned every animal egocentrically. The alignment is done by taking the nose and tail coordinates and cropping the frame to these coordinates. In order to get a tail to nose orientation from left to right we compute a rotation matrix and rotate the the resulting frame around the centre between nose and tail. This results into egocentrically aligned frames and marker coordinates $\mathbf{X} \in \mathbb{R}^{2m \times N}$, where N represents the sequence length. To fit our machine learning model we subdivide this sequence into smaller subsequences $\mathbf{x}_i$ by applying a sliding window of length T. Furthermore, we created $\tilde{\mathbf{x}}_{i+1}$ that stores the subsequent $\tilde{T}$ time points of $\mathbf{x}_i$.

For low-dimensional visualization of spatial as well as spatiotemporal data we employed Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) from the `umap-learn` Python package. All embeddings were created with the parameters `min_dist` set to 0.2 and `neighbors` set to 20.

4.3 Variational Animal Motion Embedding

Given a set of n multivariate time series $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where each time series $\mathbf{x}_i = (x_i^1, x_i^2, \ldots, x_i^T)$ contains $2m \times T$ ordered real values. The objective of our model is to learn a $d-$dimensional vector $\mathbf{z}_i \in \mathbb{R}^d$ which contains the latent representation of the input sequence $\mathbf{x}_i$. $\mathbf{z}_i$ is learned via the non-linear mappings $f_{enc} : \mathbf{x}_i \rightarrow \mathbf{z}_i$ and $f_{dec} : \mathbf{z}_i \rightarrow \tilde{\mathbf{x}}_i$, where f_{enc}, f_{dec} denotes the encoding and decoding process, respectively and is defined by,

$$\mathbf{z}_i = f_{enc}(\mathbf{x}_i)\,. \tag{1}$$

In order to learn the spatiotemporal latent representation our model encoder is parameterized by a two layer bi-directional RNN with parameters ϕ. Furthermore, our model uses two decoder, a one-directional RNN with parameters θ and a bi-directional RNN with parameter η.

As our input data is temporally dependent, RNNs are a natural choice in order to capture temporal dynamics by recursively processing each input and updating their internal state $\mathbf{h}_t$ at each timestep via,

$$\mathbf{h}_t = f_\theta(\mathbf{x}_t, \mathbf{h}_{t-1}), \tag{2}$$

where f is a deterministic non-linear transition function, and θ is the parameter set of f. The transition function f is usually modelled as long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) or gated recurrent unit (GRU) (Cho et al., 2014). Here, we use GRUs as transition function in the encoder and decoder.

The joint probability of a time series $\mathbf{x}_i$ is factorized by a RNN as product of conditionals,

$$p_\theta(\mathbf{x}_i) = \prod_{t=1}^{T} p_\theta(x_t | x_{1:t-1}). \tag{3}$$

In order to learn a joint distribution over all variables, or more precise, the underlying generative process of the data, we apply the framework of variational autoencoders (VAE) introduced by (Kingma & Welling, 2014; Rezende et al., 2014). VAEs have been shown to effectively model complex multivariate distirbutions and can generalize much better to new situations, e.g. spontaneous events, than their counterparts, discriminative models.

4.3.1 Variational Autoencoder

In brief, by introducing a set of latent random variable $\mathbf{Z}$ the VAE model is able to learn variations in the observed data and can generate $\mathbf{X}$ through conditioning on $\mathbf{Z}$. Hence, the joint probability distribution is defined as,

$$p_\theta(\mathbf{X}, \mathbf{Z}) = p_\theta(\mathbf{X}|\mathbf{Z}) p_\theta(\mathbf{Z}), \tag{4}$$

and parameterized by θ.

Obtaining the data distribution $p(\mathbf{X})$ by marginalization is intractable due to the non-linear mappings between $\mathbf{X}$ and $\mathbf{Z}$ and the integration of $\mathbf{Z}$. In order to overcome the problem of intractable posteriors the VAE framework introduces an approximation of the posterior $q_\phi(\mathbf{Z}|\mathbf{X})$ and optimizes a lower-bound on the marginal likelihood,

$$\log p_\theta(\mathbf{X}) \geq \mathbb{E}_{q_\phi(\mathbf{Z}|\mathbf{X})}[\log p_\theta(\mathbf{X}|\mathbf{Z})] - KL(q_\phi(\mathbf{Z}|\mathbf{X}) || p_\theta(\mathbf{Z})), \tag{5}$$

where $KL(Q||P)$ denotes the Kullback-Leibler divergence between two probability distributions Q and P. The prior $p_\theta(\mathbf{Z})$ and the approximate posterior $q_\phi(\mathbf{Z}|\mathbf{X})$ are typically chosen to be in a simple parametric form, such as a Gaussian distribution with diagonal covariance. The generative model $p_\theta(\mathbf{X}|\mathbf{Z})$ and the inference model $q_\phi(\mathbf{Z}|\mathbf{X})$ are trained jointly by optimzing Eq. 5 w.r.t their parameters. Using the *reparameterization trick* (Eq. 6), introduced by (Kingma & Welling, 2014) the whole model can be trained through standard backpropagation techniques for stochastic gradient descent.

4.3.2 Variational lower bound of VAME

In our case, the inference model (or encoder) $q_\phi(\mathbf{z}_i|\mathbf{x}_i)$ is parameterized by a RNN. By concatenating the last hidden states $\mathbf{h}_t$ of each layer of the encoder we obtain a global hidden state $\mathbf{h}_i$ which is a fixed-length vector representation of the entire sequence $\mathbf{x}_i$. To obtain the probabilistic latent representation $\mathbf{z}_i$ we define a prior distribution over the latent variables $p_\theta(\mathbf{z}_i)$ as an isotropic multivariate Normal distribution $\mathcal{N}(\mathbf{z}_i; \mathbf{0}, \mathbf{I})$. Its parameter μ_z and Σ_z of the approximate posterior distribution $q_\phi(\mathbf{z}_i|\mathbf{x}_i)$ are generated from the final encoder hidden state by using two fully connected layers with a Linear and a SoftPlus activation, respectively, as proposed in (?, ?). The latent representation $\mathbf{z}_i$ is then sampled from the approximate posterior and computed via the reparameterization trick,

$$\mathbf{z}_i = \mu_z + \sigma_z \odot \epsilon \quad , \tag{6}$$

where ϵ is an auxiliary noise variable and $\odot$ denotes the Hadamard product.

The generative model $p_\theta(\mathbf{x}_i|\mathbf{z}_i)$ (or decoder) receives $\mathbf{z}_i$ as input at each timestep t and aims to reconstruct $\mathbf{x}_i$. We use Mean Squared Error (MSE) as reconstruction loss, defined by,

$$\mathcal{L}_{MSE} = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \tilde{\mathbf{x}}_i||_2^2 \, . \tag{7}$$

The log-likelihood of $\mathbf{x}_i$ can be expressed as in Eq. 5. Since the KL divergence is non-negative the log-likelihhood can be written as

$$\mathcal{L}(\theta, \phi; \mathbf{x}_i) = \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_\theta(\mathbf{x}_i|\mathbf{z}_i)] - KL(q_\phi(\mathbf{z}_i|\mathbf{x}_i)||p_\theta(\mathbf{z}_i)) \, . \tag{8}$$

Here, $\mathcal{L}(\theta, \phi; \mathbf{x}_i)$ is a lower bound on the log-likelihood and therefore called the *evidence lower bound* (ELBO) as formulated by (Kingma & Welling, 2014).

We extend the ELBO in our model by an additional prediction decoder $p_\eta(\tilde{\mathbf{x}}_i|\mathbf{z}_i)$ to predict the evolution $\tilde{\mathbf{x}}_i$ of $\mathbf{x}_i$, parameterized by η. The motivation for this additional model is based on (Srivastava et al., 2015) where the authors propose a composite RNN model which aims to jointly learn important features for reconstruction and predicting subsequent video frames. Here, $p_\eta(\tilde{\mathbf{x}}_i|\mathbf{z}_i)$ serves as a regularization for learning $\mathbf{z}_i$ so that the latent representation not only memorizes an input time series but also estimates its future dynamics. Thus, we extend Eq. 8 by an additonal term and parameter,

$$\mathcal{L}(\theta, \phi, \eta; \mathbf{x}_i) = \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_\theta(\mathbf{x}_i|\mathbf{z}_i)] + \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_\eta(\tilde{\mathbf{x}}_i|\mathbf{z}_i)] - KL(q_\phi(\mathbf{z}_i|\mathbf{x}_i)||p_\theta(\mathbf{z}_i)) \, . \tag{9}$$

In order to improve the performance of the post-hoc clustering we incorporate a k-means objective based on spectral relaxation into the model as proposed by (Ma et al., 2019) to guide the learning of the network. Briefly, given a data matrix $\mathbf{z}_i \in \mathbb{R}^{d \times N}$, (Zha, He, Ding, Gu, & Simon, 2002) transformed the k-means objective into a trace maximization problem associated with the Gram matrix $\mathbf{z}_i^T \mathbf{z}_i$. Thus, the k-means objective has the form,

$$\mathcal{L}_{k-means} = Tr(\mathbf{z}_i^T \mathbf{z}_i) - Tr(\mathbf{A}^T \mathbf{z}_i^T \mathbf{z}_i \mathbf{A}) \quad , \tag{10}$$

where Tr denotes the matrix trace. $\mathbf{A} \in \mathbb{R}^{N \times k}$ is called the cluster indicator matrix and can be set to an arbitrary orthogonal matrix which further relaxes the minimization in Eq. 10 to the trace maximizaton problem

$$\max_{\mathbf{A}} Tr(\mathbf{A}^T \mathbf{z}_i^T \mathbf{z}_i \mathbf{A}), \;\; s.t. \;\; \mathbf{A}^T \mathbf{A} = \mathbf{I} \, . \tag{11}$$

Eq. 11 has a closed-form solution based on the *Ky Fan* theorem (Fan & Hoffman, 1955) which states that we need to compute the largest k eigenvectors of the Gram matrix, which gives a *lower bound* for the minimum of the k-means objective. In practice, we update $\mathbf{A}$ by computing the sum of the k-first singular values of $\sqrt{\mathbf{z}_i^T \mathbf{z}_i}$.

We are motivating this trace term as a prior on the latent vector $\mathbf{z}_i$ by assuming a joint probability of the form

$$p_\theta(\mathbf{x}_i, \mathbf{z}_i, k) = p_\theta(\mathbf{x}_i|\mathbf{z}_i, k) p_\theta(\mathbf{z}_i|k) p_\theta(k), \tag{12}$$

where $p_\theta(\mathbf{z}_i|k)$ is the trace optimisation prior. Along with this joint probability we can write

$$p_\theta(\mathbf{x}_i|\mathbf{z}_i, k) = p_\theta(\mathbf{x}_i|\mathbf{z}_i) \tag{13}$$

so that $\mathbf{x}_i$ and k are independent conditioned on $\mathbf{z}_i$. Using Bayes theorem we find that

$$p_\theta(\mathbf{z}_i|\mathbf{x}_i, k) = \frac{p_\theta(\mathbf{x}_i|\mathbf{z}_i, k) p_\theta(\mathbf{z}_i|k) p_\theta(k)}{p_\theta(\mathbf{x}_i)}, \tag{14}$$

where $p_\theta(\mathbf{z}_i|\mathbf{x}_i, k)$ is approximated by the encoder $q_\phi(\mathbf{z}_i|\mathbf{x}_i, k)$.

Using Jensen's inequality, the log-likelihood of VAME can be written as:

$$\log p(\mathbf{x}_i) = \log \int_{\mathbf{z}_i} \sum_k p_\theta(\mathbf{x}_i, \mathbf{z}_i, k) d\mathbf{z}_i \geq E_{q(\mathbf{z}_i|\mathbf{x}_i,k)}[\log \frac{p_\theta(\mathbf{x}_i, \mathbf{z}_i, k)}{q(\mathbf{z}_i|\mathbf{x}_i, k)}] = \mathcal{L}_{ELBO}(\mathbf{x}_i). \tag{15}$$

We can now express the lower bound on the log-likelihood with an additional prior on the latent vector in the form of

$$\mathcal{L}(\theta, \phi, \eta; \mathbf{x}_i) = \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_\theta(\mathbf{x}_i|\mathbf{z}_i) + \log p_\eta(\tilde{\mathbf{x}}_i|\mathbf{z}_i)] - KL(q_\phi(\mathbf{z}_i|\mathbf{x}_i, k)||p_\theta(\mathbf{z}_i|k)) \,. \tag{16}$$

As stated by (Ma et al., 2019), $\mathbf{z}_i$ is learned by the model and therefore not static. Therefore, Eq. 10 can be regarded as a regularization term for learning $\mathbf{z}_i$. Note that the balance between Eq. 10 and Eq. 9 forces the encoder to learn a more defined cluster boundary. Finally, the training objective to minimize is

$$\min_{\theta,\phi,\eta} \mathcal{L}(\theta, \phi, \eta; \mathbf{x}_i) \tag{17}$$

and the overall loss function can be written as

$$\mathcal{L}_{total} = \mathcal{L}_{reconstruction} + \mathcal{L}_{prediction} + \mathcal{L}_{KL} + \mathcal{L}_{k-means} \quad , \tag{18}$$

where $\mathcal{L}_{prediction}$ is the MSE loss of the prediction decoder.

The full model was trained on the combined dataset ($1.3e6$ time points) using the Adam optimizer (Kingma & Ba, 2015) with a fixed learning rate of 0.0005 on a single Nvidia 1080ti GPU. All computing was done with PyTorch (Paszke et al., 2017). The training error for all loss-terms employed in (18) for the data presented in the Results section is plotted in Figure S.7. The ergodic mean of the reconstruction error $\mathcal{L}_{reconstruction}$ for all virtual marker time series was found to be 1.82 pixels.

4.4 Clustering into behavioral motifs

To determine the set of behavioral motifs $B = \{b_1, \ldots, b_K\}$ we first obtained the latent vector $\mathbf{z}$ from a given dataset using the machine learning framework described in Methods 4.3. Given the pose tracking yields d time series extracted from a video containing N frames and the size of the spatiotemporal time window is T, the resulting feature matrix $\mathcal{F}$ is of dimensionality $d \times (N - T)$. We then performed k-means clustering on $\mathcal{F}$ to identify K behavioral motifs. Figure 1 (C, Middle) and Figure S.1 show exemplary state sequences obtained from the clustering.

We can then determine the motif usage as the percentage of video frames that are assigned to the occurrence of a specific motif. Furthermore we may model the transitions between behavioral motifs as a discrete-time Markov chain where the transition probability into a future motif is only dependent on the present motif. This results in a $K \times K$ transition probability matrix $\mathcal{T}$, with the elements

$$\mathcal{T}_{lk} = P(b_k|b_l), \tag{19}$$

being the transition probabilities from one motif $b_l \in B$ to another motif $b_k \in B$, that are empirically estimated from clustering of $\mathcal{F}$.

Next we can compute the stationary distribution of the Markov chain $\mathcal{T}$, which is a probability distribution to which the chain converges when time progresses. By definition, the stationary distribution π satisfies

$$\pi = \pi\mathcal{T}. \tag{20}$$

In other words, π is invariant by the matrix $\mathcal{T}$.

In order to obtain a hierarchical representation of behavioral motifs we can represent the Markov chain (19) as a directed graph $\mathbb{G}$ consisting of nodes $v_1 \ldots v_K$ connected by edges with an assigned transition probability $\mathcal{T}_{lk}$. Additionally, the size of each node corresponds to the total occurrence of the behavior motif throughout all N video frames. We can the transform $\mathbb{G}$ into a binary tree $\mathbb{T}$ by iteratively merging two nodes (v_i, v_j) until only the root node v_R is left. To select i and j in each reduction step, we compute the cost function

$$C_R = \min_{i,j} \left(\sum_{i,j} \frac{U_i + U_j}{\mathcal{T}_{ij} + \mathcal{T}_{ji}} \right), \tag{21}$$

where U_i is the probability of occurrence for the ith motif. Note that after each reduction step the matrix $\mathcal{T}$ is recomputed in order to account for the merging of nodes.

Lastly, we may obtain *communities* of behavioral motifs by cutting $\mathcal{T}$ at given depth of the tree, analogous to the hierarchical clustering approach used for dendrograms.

4.5 Manually assigned labels and scoring

In order to obtain manually assigned labels of behavioral motifs we asked three experts to annotate one recording of freely moving behavior with a duration of 6 minutes. All three experts had a strong experience with in-vivo experiments as well as ethogram-based behavior quantification. The experts could scroll trough the video in slow-motion forward and backward in time and annotated the behavior into several atomic motifs as well as a composition of those. As an example, the experts were allowed to annotate a behavioral sequence as *walk* or *exploration*, but also *walk and exploration*. We then summarized the annotation into atomic motifs into 5 coarse behavioral labels, as shown in Table 1.

Coarse label	Assigned atomic motif
Walk	Walk, walk and bend, walk and sniff
Pause	No locomotion, Bending, looking up or down while standing still
Groom	Groom
Rear	Rear, low-rear, wall-rear
Exploratory	Undirected sniffing while standing still, bending, looking up or down

Table 1: Assignment of atomic motifs into coarse behavior labels.

The coarse labels were created with respect to the behavior descriptions taken from the Mouse Ethogram database (`www.mousebehavior.org`), which provides a consensus of several previously published ethograms. The assignment of coarse labels to the Mouse Ethogram database taxonomy is shown in Table 2.

Coarse label	Mouse Ethogram database
Walk	Active behavior - General activity - Exploratory behavior - Search - General locomotion
Pause	Inactive behavior- Still and alert
Groom	Active behavior - Maintenance behaviors - Grooming
Rear	Active behavior - General activity - Exploratory behavior - Search - Rearing
Exploratory	Active behavior - General activity - Exploratory behavior - Investigate - Undirected sniffing

Table 2: Mouse Ethology database taxonomy corresponding for each manually assigned coarse label.

For scoring of human assigned labels to the behavioral to VAME motifs we used the clustering evaluation measure Purity and NMI. Purity is defined as

$$\text{Purity}(U,V) = \frac{1}{N}\sum_{u\in U}\max_{v\in V}|u\cap v|, \tag{22}$$

where U it the set of manually assigned labels u, V is the set of labels generated by VAME v and N is the number of frames in the behavioral video. The Normalized Mutual Information score is written as

$$\text{NMI}(U,V) = \frac{\text{MI}(U,V)}{E(H(U),H(V))}, \tag{23}$$

where $\text{MI}(U,V)$ is the mutual information between set U and V defined as

$$\text{MI}(U,V) = \sum_{u\in U}\sum_{v\in V}\frac{|u\cap v|}{N}\log\left(\frac{N|u\cap v|}{|u||v|}\right), \tag{24}$$

and $H(U)$ is the entropy of set U defined as

$$H(U) = -\sum_{i=1}^{|U|}P_i\log(P_i), \tag{25}$$

where P_i now denotes the probability for the ith entry of U.

Note that the Purity score (22) is larger when the set V is larger than U and the NMI score (23) is generally larger when both sets U and V are of similar size, i.e. the number of possible labels is roughly the same in the human assigned set as well as the set generated using VAME.

4.6 Human phenotype classification task

For the classification of phenotypes using human experts we have created an online form, where experts could watch all behavior videos and make their choice about which phenotype is shown for each video. The average time to complete

the questionnaire for N=8 animals was 30 minutes. The participants have not been told how many animals of each group are in the set. For every video, the following five decision could be made: *APP/PS1 (Very sure), APP/PS1 (Likely), Unsure, Wildtype (Likely), Wildtype (Very Sure)*. However, we have counted each of right answers (Very sure and Likely) as a correct classification (1 point), and both wrong answers as well as the choice for the Unsure option as wrong classification (0 points). Note that we did not give a time limit but asked for the time they spend on the task. In average, every participant spent around 30 min to complete the task. We had eleven experts participating in this classification task. All of them had previous experience with behavioral video recordings in an open field and/or treadmill setting. In addition, six of the participants also had previous experience with the APP/PS1 phenotype. In the end of the task we asked how they tried to identify the APP/PS1 phenotype to uncover if there is a certain strategy human experts are sharing.

4.7 Code availability

The VAME toolbox is available at `https://github.com/LINCellularNeuroscience/VAME`.

5 Acknowledgments

We thank J. Macke, E. Restrepo, J. Gall and S. Stober for comments on the manuscript. This work was supported by the European Research Council (CoG;SUBDECODE) and DFG-SFB 1089.

References

Anderson, D. J., & Perona, P. (2014). Toward a Science of Computational Ethology. *Neuron*, *84*(1), 18–31. doi: 10.1016/j.neuron.2014.09.005

Baek, J.-H., Cosman, P., Feng, Z., Silver, J., & Schafer, W. R. (2002). Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *Journal of Neuroscience Methods*, *118*(1), 9–21. doi: 10.1016/S0165-0270(02)00117-6

Batty, E., Whiteway, M., Saxena, S., Biderman, D., Abe, T., Musall, S., ... Paninski, L. (2019). BehaveNet: nonlinear embedding and Bayesian neural decoding of behavioral videos. In *Advances in Neural Information Processing Systems 32* (pp. 15680–15691).

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798-1828.

Berman, G. J. (2018). Measuring behavior across scales. *BMC biology*, *16*(1), 23.

Berman, G. J., Choi, D. M., Bialek, W., & Shaevitz, J. W. (2014). Mapping the stereotyped behaviour of freely moving fruit flies. *Journal of The Royal Society Interface*, *11*(99), 20140672. doi: 10.1098/rsif.2014.0672

Brown, A. E. X., & de Bivort, B. (2018). Ethology as a physical science. *Nature Physics*, *14*(7), 653–657. doi: https://doi.org/10.1038/s41567-018-0093-0

Cande, J., Namiki, S., Qiu, J., Korff, W., Card, G. M., Shaevitz, J. W., ... Berman, G. J. (2018, June). Optogenetic dissection of descending behavioral control in Drosophila. *eLife*, *7*, e34275. Retrieved 2020-10-22, from `https://doi.org/10.7554/eLife.34275` (Publisher: eLife Sciences Publications, Ltd) doi: 10.7554/eLife.34275

Carew, T. J. (2005). *Behavioral neurobiology: the cellular organization of natural behavior*. Oxford University Press, Incorporated, 2000.

Chang, S., Zhang, Y., Han, W., Yu, M., Guo, X., Tan, W., ... Huang, T. S. (2017). Dilated recurrent neural networks. In *Advances in neural information processing systems 30* (pp. 77–87).

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems 29* (pp. 2172–2180).

Cho, K., van Merriënboer, B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., & 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* (pp. 1724–1734).

Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. C., & Bengio, Y. (2015). A recurrent latent variable model for sequential data. In *Advances in neural information processing systems 28* (pp. 2980–2988).

Crawley, J. N. (2008). Behavioral phenotyping strategies for mutant mice. *Neuron*, *57*(6), 809–818. doi: 10.1016/j.neuron.2008.03.001

Datta, S. R., Anderson, D. J., Branson, K., Perona, P., & Leifer, A. (2019). Computational Neuroethology: A Call to Action. *Neuron*, *104*(1), 11–24. doi: 10.1016/j.neuron.2019.09.038

Fan, K., & Hoffman, A. J. (1955). Some metric inequalities in the space of matrices. *Proceedings of the American Mathematical Society*, *6*(1), 111–116.

Fox, E. B., Sudderth, E. B., Jordan, M. I., & Willsky, A. S. (2011). A sticky HDP-HMM with application to speaker diarization. *The Annals of Applied Statistics*, *5*(2A), 1020–1056. doi: 10.1214/10-AOAS395

Gomez-Marin, A., Paton, J. J., Kampff, A. R., Costa, R. M., & Mainen, Z. F. (2014). Big behavioral data: psychology, ethology and the foundations of neuroscience. *Nature Neuroscience*, *17*(11), 1455–1462. doi: https://doi.org/10.1038/nn.3812

Günel, S., Rhodin, H., Morales, D., Campagnolo, J. a., Ramdya, P., & Fua, P. (2019). DeepFly3D, a deep learning-based approach for 3D limb and appendage tracking in tethered, adult Drosophila. *eLife*, *8*, e48571. doi: 10.7554/eLife.48571

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M. M., ... Lerchner, A. (2017). beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th international conference on learning representations, ICLR*.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735–1780.

Huang, H., Nie, S., Cao, M., Marshall, C., Gao, J., Xiao, N., ... Xiao, M. (2016). Characterization of AD-like phenotype in aged APPSwe/PS1dE9 mice. *Age (Dordrecht, Netherlands)*, *38*(4), 303–322. doi: 10.1007/s11357-016-9929-7

Jankowsky, J. L., Fadale, D. J., Anderson, J., Xu, G. M., Gonzales, V., Jenkins, N. A., ... Borchelt, D. R. (2004). Mutant presenilins specifically elevate the levels of the 42 residue beta-amyloid peptide in vivo: evidence for augmentation of a 42-specific gamma secretase. *Human Molecular Genetics*, *13*(2), 159–170. doi: 10.1093/hmg/ddh019

Jankowsky, J. L., Slunt, H. H., Ratovitski, T., Jenkins, N. A., Copeland, N. G., & Borchelt, D. R. (2001). Co-expression of multiple transgenes in mouse CNS: a comparison of strategies. *Biomolecular Engineering*, *17*(6), 157–165. doi: 10.1016/S1389-0344(01)00067-3

Janus, C., Flores, A. Y., Xu, G., & Borchelt, D. R. (2015). Behavioral abnormalities in APPSwe/PS1dE9 mouse model of AD-like pathology: comparative analysis across multiple behavioral domains. *Neurobiology of Aging*, *36*(9), 2519–2532. doi: 10.1016/j.neurobiolaging.2015.05.010

Jiang, Z., Zheng, Y., Tan, H., Tang, B., & Zhou, H. (2017). Variational deep embedding: An unsupervised and generative approach to clustering. In *Proceedings of the 26th international joint conference on artificial intelligence* (p. 1965–1972).

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd international conference on learning representations, ICLR*.

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational bayes. In *2nd international conference on learning representations, ICLR*.

Kingma, D. P., & Welling, M. (2019). An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, *12*(4), 307-392. Retrieved from `http://dx.doi.org/10.1561/2200000056` doi: 10.1561/2200000056

Kobak, D., & Berens, P. (2019). The art of using t-SNE for single-cell transcriptomics. *Nature Communications*, *10*(1), 1–14. doi: 10.1038/s41467-019-13056-x

Krakauer, J. W., Ghazanfar, A. A., Gomez-Marin, A., MacIver, M. A., & Poeppel, D. (2017). Neuroscience Needs Behavior: Correcting a Reductionist Bias. *Neuron*, *93*(3), 480–490. doi: 10.1016/j.neuron.2016.12.041

Kuehne, H., Richard, A., & Gall, J. (2020, 12). A hybrid rnn-hmm approach for weakly supervised temporal action segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*, 765-779. doi: 10.1109/TPAMI.2018.2884469

Lalonde, R., Kim, H. D., & Fukuchi, K. (2004). Exploratory activity, anxiety, and motor coordination in bigenic APPswe + PS1/DeltaE9 mice. *Neuroscience Letters*, *369*(2), 156–161. doi: 10.1016/j.neulet.2004.07.069

Luxem, K., Fuhrmann, F., Remy, S., & Bauer, P. (2019). Hierarchical network analysis of behavior and neuronal population activity. In *2019 Conference on Cognitive Computational Neuroscience*. Berlin, Germany: Cognitive Computational Neuroscience. doi: 10.32470/CCN.2019.1261-0

Ma, Q., Zheng, J., Li, S., & Cottrell, G. W. (2019). Learning representations for time series clustering. In *Advances in neural information processing systems 32* (pp. 3776–3786).

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. New York: Cambridge University Press.

Markowitz, J. E., Gillis, W. F., Beron, C. C., Neufeld, S. Q., Robertson, K., Bhagat, N. D., ... Datta, S. R. (2018). The Striatum Organizes 3d Behavior via Moment-to-Moment Action Selection. *Cell*, *174*(1), 44–58.e17.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., & Bethge, M. (2018). DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience*, *21*(9), 1281–1289.

McIlwain, K. L., Merriweather, M. Y., Yuva-Paylor, L. A., & Paylor, R. (2001). The use of behavioral test batteries: effects of training history. *Physiology & Behavior*, *73*(5), 705–717. doi: 10.1016/s0031-9384(01)00528-5

Musall, S., Kaufman, M. T., Juavinett, A. L., Gluf, S., & Churchland, A. K. (2019). Single-trial neural dynamics are dominated by richly varied movements. *Nat Neurosci*, *22*(10), 1677–1686. doi: 10.1038/s41593-019-0502-4

Onos, K. D., Uyar, A., Keezer, K. J., Jackson, H. M., Preuss, C., Acklin, C. J., ... Howell, G. R. (2019). Enhancing face validity of mouse models of Alzheimer's disease with natural genetic variation. *PLoS genetics*, *15*(5), e1008155. doi: 10.1371/journal.pgen.1008155

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., ... Sussillo, D. (2018). Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, *15*(10), 805–815. doi: 10.1038/s41592-018-0109-9

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... Lerer, A. (2017). Automatic differentiation in PyTorch. In *NIPS autodiff workshop.*

Pereira, J., & Silveira, M. (2019). Learning Representations from Healthcare Time Series Data for Unsupervised Anomaly Detection. In *2019 IEEE International Conference on Big Data and Smart Computing (BigComp)* (pp. 1–7). (ISSN: 2375-933X) doi: 10.1109/BIGCOMP.2019.8679157

Pereira, T. D., Aldarondo, D. E., Willmore, L., Kislin, M., Wang, S. S.-H., Murthy, M., & Shaevitz, J. W. (2019). Fast animal pose estimation using deep neural networks. *Nature Methods*, *16*(1), 117–125. doi: 10.1038/s41592-018-0234-5

Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st international conference on machine learning* (Vol. 32, pp. 1278–1286).

Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2019). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, S0169207019301888. doi: 10.1016/j.ijforecast.2019.07.001

Speiser, A., Yan, J., Archer, E. W., Buesing, L., Turaga, S. C., & Macke, J. H. (2017). Fast amortized inference of neural activity from calcium imaging data with variational autoencoders. In *Advances in neural information processing systems 30* (pp. 4024–4034).

Srivastava, N., Mansimov, E., & Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In F. Bach & D. Blei (Eds.), *Proceedings of the 32nd international conference on machine learning* (Vol. 37, pp. 843–852). PMLR.

Tinbergen, N. (1951). *The study of instinct*. Clarendon Press.

Wiltschko, A. B., Johnson, M. J., Iurilli, G., Peterson, R. E., Katon, J. M., Pashkovski, S. L., ... Datta, S. R. (2015). Mapping Sub-Second Structure in Mouse Behavior. *Neuron*, *88*(6), 1121–1135.

Zha, H., He, X., Ding, C., Gu, M., & Simon, H. D. (2002). Spectral relaxation for k-means clustering. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems 14* (pp. 1057–1064). MIT Press.

Supplemental Materials

5.1 Absolute numbers and model comparison

k = 15	Abs. Purity	Abs. NMI	Rel. Purity %	Rel. NMI %
Baseline (Spatial signal)	70.18	19.58	-	-
Spatio-temporal signal (SVD)	69.53	19.87	-	1.48
Spatio-temporal signal (AR-HMM)	73.07	22.87	4.18	16.8
Spatio-temporal signal (VAME)	74.83	26.85	6.63	37.13
Spatio-temporal + prediction (VAME)	**75.11**	**27.44**	**7.03**	**40.14**

k = 30	Abs. Purity	Abs. NMI	Rel. Purity %	Rel. NMI %
Baseline (Spatial signal)	71.1	18.97	-	-
Spatio-temporal signal (SVD)	71.1	20.04	-	5.64
Spatio-temporal signal (AR-HMM)	73.58	23.44	3.49	23.56
Spatio-temporal signal (VAME)	77.19	26.86	8.57	41.59
Spatio-temporal + prediction (VAME)	**77.2**	**27.93**	**8.58**	**47.23**

k = 45	Abs. Purity	Abs. NMI	Rel. Purity %	Rel. NMI %
Baseline (Spatial signal)	71.6	18.85	-	-
Spatio-temporal signal (SVD)	71.3	19.62	-	4.09
Spatio-temporal signal (AR-HMM)	73.75	23.68	3	25.62
Spatio-temporal signal (VAME)	77.63	**27.15**	8.42	**44.03**
Spatio-temporal + prediction (VAME)	**77.77**	27	**8.62**	43.24

Table S.1: Absolute values and model comparison

k = 17	Abs. Purity	Abs. NMI	Rel. Purity %	Rel. NMI %
Baseline (Spatial signal)	70.03	18.1	-	-
MotionMapper	72.96	23.22	4.18	28.29
Spatio-temporal + prediction (VAME)	**75.12**	**25.78**	**7.27**	**42.43**

Table S.2: Comparison to the MotionMapper framework

Absolute values for the scoring with manually annotated as shown in Figure 5 are shown in Table S.1. Furthermore, we have compared the performance of VAME against Singular Value Decomposition (SVD), a linear dimension reduction method that is closely related to Principal Component Analysis (PCA). For this purpose we have obtained the first 12 singular values computed for the identical time window T, that was otherwise fed to VAME. The singular values explained more than 95% of the original input data. We have then clustered the singular values for each time window using k-Means and computed the Purity and NMI score, shown in Table S.1.

For comparison of our model to the AR-HMM we employed the original codebase supplied by the authors (Wiltschko et al., 2015). We used default parameter settings for all values ($\gamma = 999$, $N_{lags} = 3$, $\nu = 4$) while the maximum number of states was set to the corresponding cluster size k. The *sticky* parameter setting was employed and the value of κ has been set to the number of datapoints, as suggested by the authors [2].

To compare our model to the *MotionMapper* framework (Berman et al., 2014), we used the original codebase provided by the authors. The comparison has been carried out on the validation dataset discussed in Figure 5, that consists of 20.000 datapoints sampled at 60 Hz. We have first converted the 12-dimensional input signal to the time-frequency domain using the Wavelet transform (15 frequency bins in the range between 0 and 30 Hz). For the stacked spectrogram we then obtained a two dimensional t-SNE embedding (perplexity=32, learning rate=200, 3000 iterations). The watershed segmentation of the embedding space then yielded 17 separate regions, that were compared as discrete labels to the human annotated dataset. The obtained Purity and NMI measures are shown in Table S.2.

[2] According to the usage documentation available within the MoSeq repository.

5.2 Parameters and biases

Although the proposed behavior quantification method is unsupervised, the choice of many parameters as well as processing steps integrate biases into the quantification result. In Table S.3 we summarize the biases of our approach and state the parameter setting that have been used for our data.

Choice	Our setting
What is the pixel resolution of the animal body in the camera image	In our case, the animal body was covered with approximately 100×300 pixels.
What is the temporal resolution of the camera?	The employed camera operates at 60 Hz.
How many virtual markers are used and how are they placed on the animal body parts?	We placed 6 virtual markers on paws, the nose and the tail root (see Figure 1).
Which software is used for keypoint detection?	We used DeepLabCut 1.1 (Mathis et al., 2018).
How many frames are manually labeled?	We labeled the keypoints in 600 uniformly chosen frames.
How long is the keypoint detection CNN trained?	We trained the keypoint detection for 350.000 iterations, the resulting training errors was 2.14 pixels and the test error was 2.51 pixels.
How is the obtained keypoint time series aligned to egocentric coordinates?	We use the procedure defined in the Methods section.
How is missing data handled?	Missing data points for periods where pose estimation could not reliably detect the position of the corresponding virtual marker shorter then 200 ms were linearly interpolated while the values for longer periods were set to an arbitrary negative value.
Which size of the time window was used for the reconstruction and prediction decoder?	We used 30 frames (500 ms) for the reconstruction and 10 frames (166 ms) for the prediction decoder.
Which other hyperarameters have been used to train VAME?	All other hyperparameter settings can be found in the default configuration file available at the project website.
How many datapoints are used to train VAME?	We trained our model with 1.3 million data points.
How is the clustering carried out?	We used k-Means clustering as described in the Methods section.

Table S.3: Table of choices for our approach and settings for our model.

5.3 Supplementary Figures

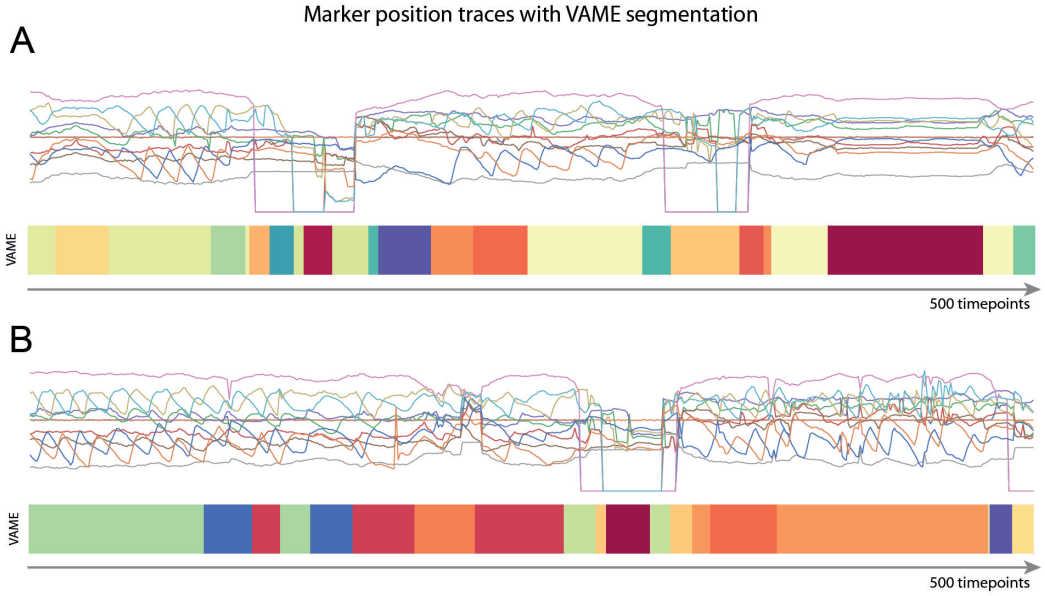

Figure S.1: **Marker x-y-position traces with VAME Segmentation.** (A, B) (Top) x-y-position of virtual markers in the open field setup. (Bottom) VAME segmentation of the given sequence snippet into discrete motifs. Note that values of the input time series were set to an arbitrary negative value if the used pose estimation tools could not reliably detect the position of the corresponding virtual marker.

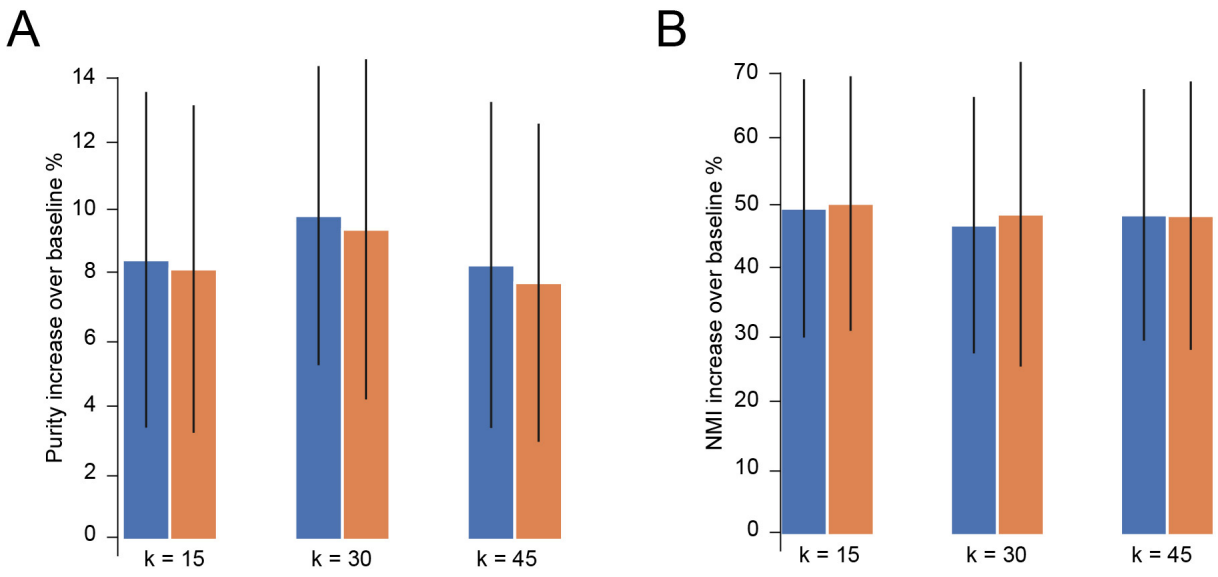

Figure S.2: **Bootstrapped Purity an NMI values for 1 minute long behavior observations.** (A) Mean Purity increase over baseline for clustering into 15, 30 and 45 motifs for 1 minute long behavioral measurements (N_{shuffle}=1000). Error bars denote the bootstrap estimate of the standard deivation over the number of shuffles. (B) Same as in (A), but for the relative NMI increase.

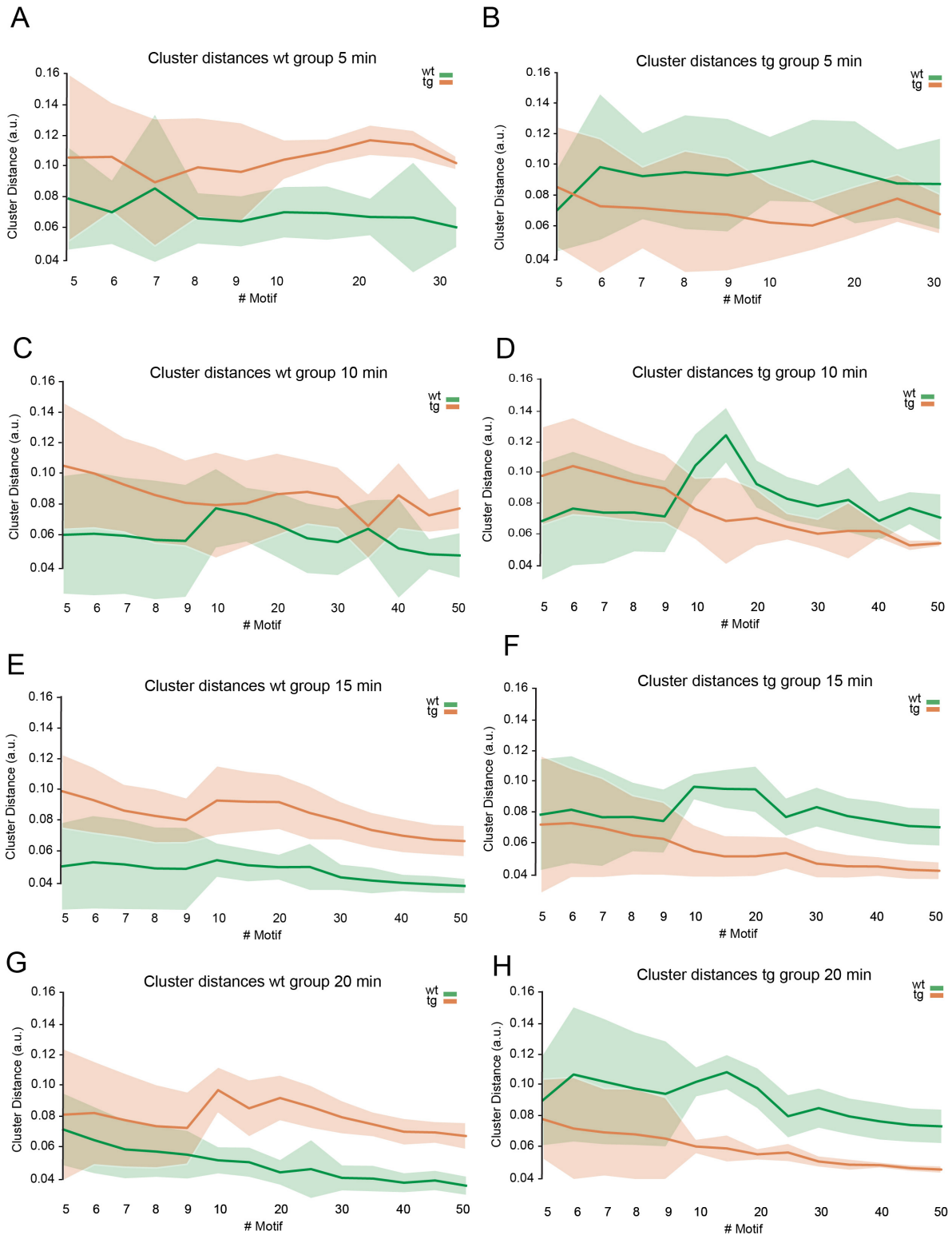

Figure S.3: **Cluster distances over different number of motifs and recording durations.** (A, B) Cluster distances for the first 5 min of recording. (C, D) Cluster distances for the first 10 minutes of recording. (E, F) Cluster distances for the first 15 minutes of recording. (G, H) Cluster distances for the first 20 minutes of recording.

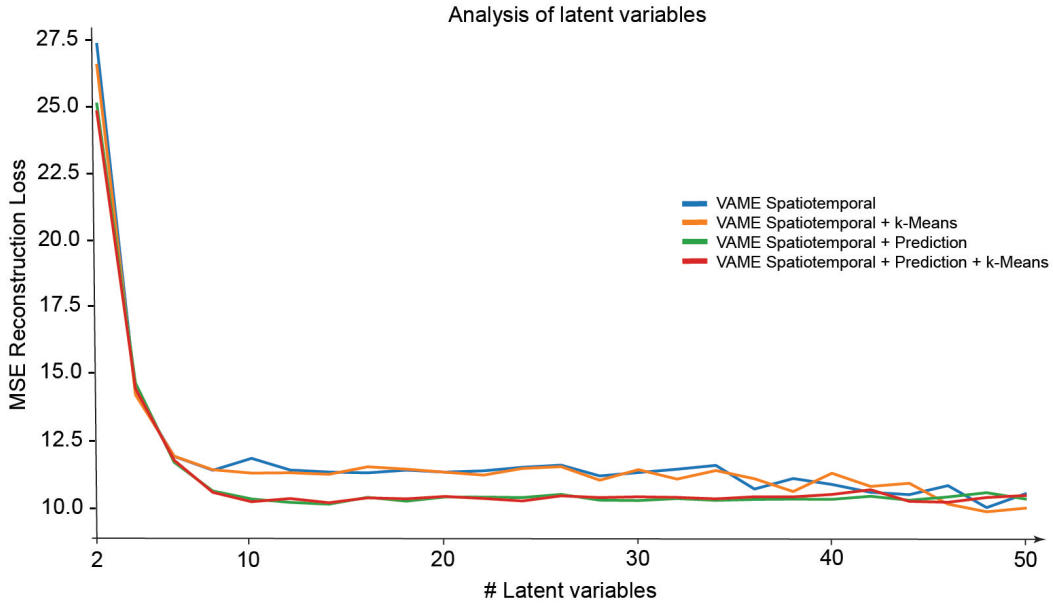

Figure S.4: **Reconstruction error after 100 epochs of training over an increasing number of latent variables.**

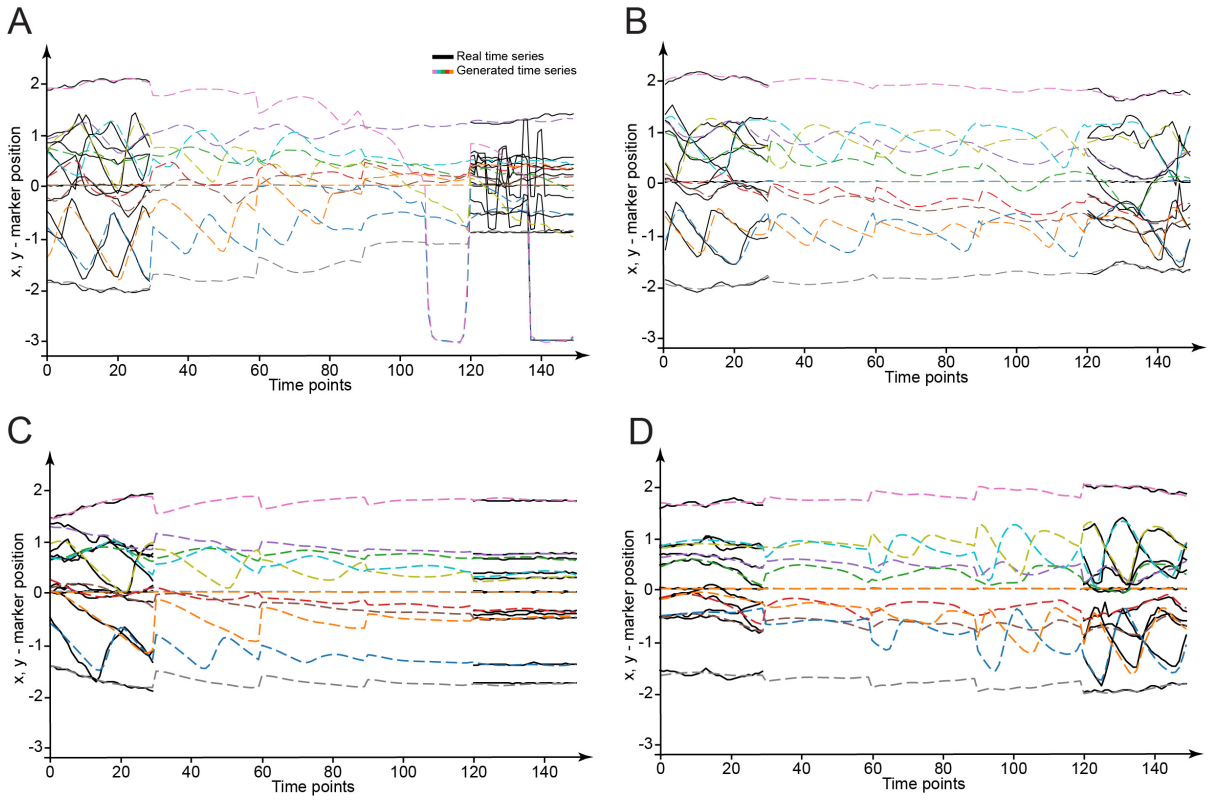

Figure S.5: **Generated time series between to latent vectors.** (A-D) Synthetic behavioral time series (colored lines) generated via latent interpolation between a given start vector $\mathbf{z}_s$ and end vector $\mathbf{z}_e$ (black lines).

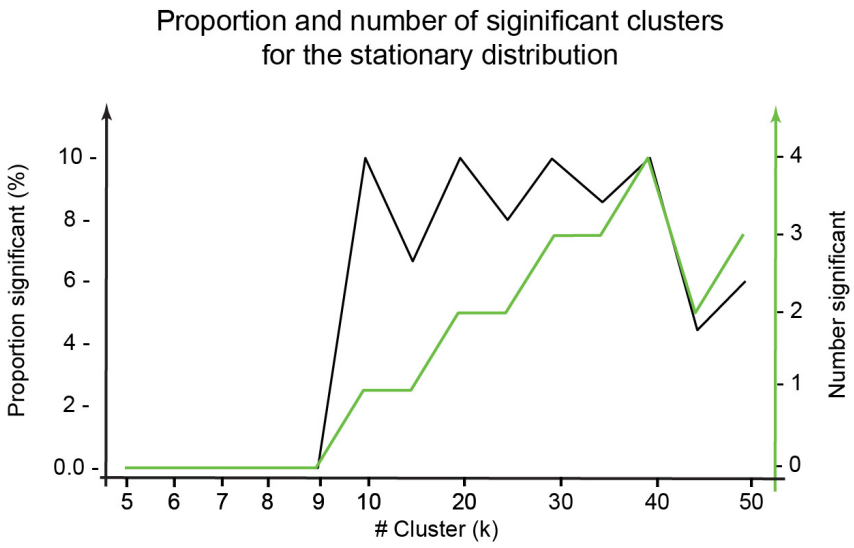

Figure S.6: **Number and proportion of motifs which are significantly changed within the stationary distribution given the cluster size k.**

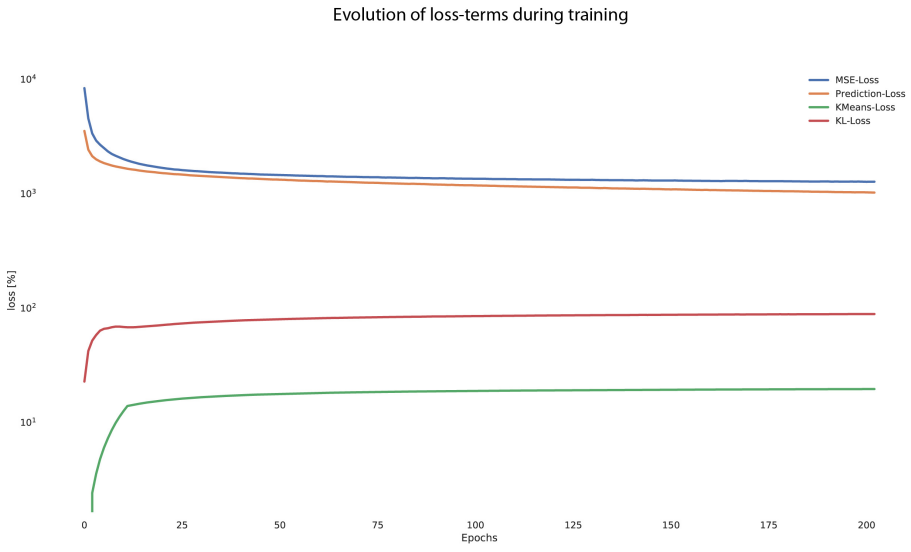

Figure S.7: **Training error for all loss-terms employed in the model.**