Probabilistic representations fail to emerge in task-optimized neural networks

Abstract

While mounting evidence indicates that human decision-making follows Bayesian principles, the underlying neural computations remain unknown. Recent work proposes that probabilistic representations arise naturally in neural networks trained with non-probabilistic objectives (Orhan & Ma, 2017). However, prior analyses did not explicitly examine whether the neural code merely re-represents inputs or performs useful transformations that prioritize three criteria for a probabilistic representation: generalization, invariance, and representational simplicity (Walker et al., 2023; Pohl et al., 2024). Using a novel probing-based approach, we show that training feed-forward networks to perform cue combination and coordinate transformation without probabilistic objectives leads to Bayesian posteriors being decodable from their hidden layer activities. However, we also show that these networks fail the generalization, invariance and representational simplicity criteria: they do not generalize out-of-sample, compress their inputs, or develop easily decodable representations. Therefore, it remains an open question under what conditions truly probabilistic representations emerge in neural networks.

Keywords: information bottleneck; probabilistic neural coding; ReLU networks; uncertainty

Background

In an uncertain world, the best way to make decisions is by using the rules of probabilistic inference; experimental evidence suggests that humans make perceptual decisions in exactly this way (Ernst & Banks, 2002; Kording & Wolpert, 2004). Competing theories about how the brain implements such inference strategies propose that neural activities either represent a parametric form of a probability distribution (e.g., probabilistic population codes, distributed distributional codes) or represent samples from it (i.e., neural sampling codes) (Haefner, Beck, Savin, Salmasi, & Pitkow, 2024). However, testing these theories in neural data and distinguishing between their disparate predictions remains an ongoing challenge.

Recent work has suggested that neural networks develop internal representations of posteriors even without explicit probabilistic objectives (Orhan & Ma, 2017). This would suggest that probabilistic representations emerge naturally when learning to behave optimally under uncertainty. However, previous decoding approaches (Orhan & Ma, 2017; Walker, Cotton, Ma, & Tolias, 2020) only assessed whether posteriors were decodable from neural activity, i.e., the *specificity* of the neural code, without testing whether these representations filter irrelevant input information, i.e., *invariance*. Thus, these methods could not distinguish truly probabilistic representations from trivial re-representations of inputs. Here, we formalize this distinction through a novel information-bottleneck (Tishby, Pereira, & Bialek, 2000) requirement: a network's internal representation should maximize information about relevant posteriors while minimizing information about inputs. The underlying insight is that task-relevant posteriors should be decodable from the hidden layer activities of a network that uses posterior uncertainty to behave optimally, but if inputs are also decodable, the network is not meaningfully transforming those inputs into a usable code for downstream computations. Additionally, Orhan and Ma (2017) tested generalization in a limited capacity: while their networks pass Bayesian transfer in an interpolation setup, they crucially do not examine out-of-sample extrapolation, which we investigate herein. Finally, we assess the representational simplicity of these probabilistic representations to determine whether flexible networks form internal representations that improve downstream decodability.

Methods

Following Orhan and Ma (2017), we trained "performer" networks to optimally perform either cue combination or coordinate transformation without probabilistic objectives. Networks consisted of an input layer divided into two populations, each with 50 independent Poisson neurons that had Gaussian tuning curves whose gain g_i was population-dependent and varied trial-by-trial, a single hidden layer with 200 neurons (and ReLU activations), and a final linear readout "layer" of a single neuron. The activities of the input populations constituted the observations ("cues") based on which the networks needed to compute their outputs. In cue combination, both populations were driven by the same latent stimulus s, which the network had to estimate based on the input layer activities. In coordinate transformation, each population was driven by a different s_i and the network had to optimally estimate the sum of the two latent stimuli $s_T = s_1 + s_2$ – a more difficult task considering the network must marginalize out s_1 and s_2 (Ma, Beck, Latham, & Pouget, 2006; Beck, Latham, & Pouget, 2011). The gain of the input populations was considered a nuisance variable that the networks had to marginalize out. We also constructed two control networks, one that trivially copied inputs to the hidden layer (COPY) and one that constructed a probabilistic population code (PPC) representation in its hidden layer. In these control networks, only hidden-to-output weights were trained.

Training Performers were trained with mean squared error loss and stochastic gradient descent (Adam optimizer). During training and testing, we varied the degree of "Bayesian transfer" (Orhan & Ma, 2017). In the "all gains" condition, networks were trained and tested on all gain combina-

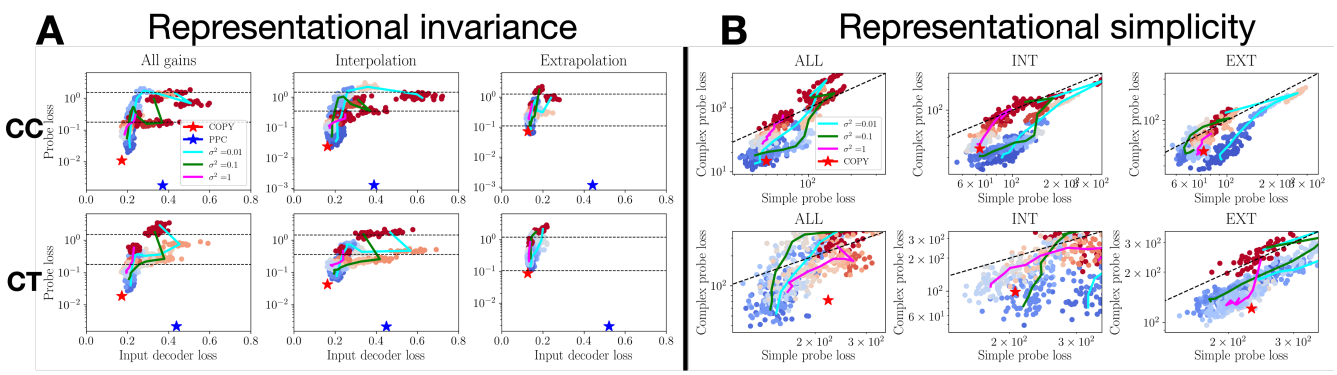

Figure 1: **A)** Information plane analysis for both tasks (**C**ue **C**ombination and **C**oordinate **T**ransformation) and training conditions (values of σ^2). In each panel, learning trajectories are plotted for all three weight initializations, and scatter plots represent test batches at different stages of learning; scatter plots are colored according to the network performance at that stage of learning (in Frac. RMSE). We show two horizontal calibration curves, the top representing the amount of information in the prior and the bottom representing the performance of a suboptimal network that does not explicitly encode uncertainty but instead learns to shift a posterior of *fixed* width to match the posterior mean. **B)** Representational simplicity analysis for all tasks and training conditions. Panels are colored similarly to A). Dashed line represents unity of simple and complex decoder performance.

tions (g_1, g_2) with $g_1, g_2 \in \mathcal{G} = \{0.25, 0.5, 0.75, 1, 1.25\}$, as in Orhan and Ma (2017). For the "Bayesian interpolation" and "Bayesian extrapolation" conditions, networks were trained on a subset of $\mathcal{G}$ and tested on the remainder of the set: $g_1, g_2 \in \{0.25, 1.25\}$ and $g_1, g_2 \in \{0.25, 0.5\}$, respectively.

We also varied the degree of richness in the training dynamics (Flesch, Juechems, Dumbalska, Saxe, & Summerfield, 2021; Farrell, Recanatesi, & Shea-Brown, 2023) by initializing all networks with zero-mean Gaussian-distributed weights and varying the variance $\sigma^2 \in \{0.01, 0.1, 1\}$. Richer learning corresponds to well-structured representational learning and occurs when network weight initializations are small, whereas lazy learning leads to less structured, high-dimensional representations and occurs for relatively large weight initializations.

We trained (also with Adam) two types of "interpreter" networks – posterior probes and input decoders – to assess the structure of the internal representations in the performer networks. Posterior probes were 2-layer ReLU networks trained (with a KL divergence loss) to decode a discretized version of the ground truth posterior distribution (which was analytically computable from input layer activations) from the performer's hidden layer activations. We also trained a second kind of probes to determine whether hidden layer activations specifically represented uncertainty in an easily decodable format. Specifically, these probes were trained (with a mean squared error loss) to simply decode the posterior variance (a scalar) from the performer's hidden layer activations, rather than the full posterior. This allowed us to compare the performance of a simple linear and a nonlinear variance probe; the linear probe had no hidden layer, while the nonlinear probe had two 200-neuron hidden layers with ReLU activations. Finally, input decoders were high-capacity ReLU networks (two 300-neuron hidden layers) trained to reconstruct the performer's input layer activations from its hidden layer activations, and they were trained with negative log-likelihood loss.

Results

Our information-bottleneck analysis reveals that none of the performer models, regardless of task or training conditions, compress information adequately, especially compared to an optimal PPC representation of the same information (Figure 1A); in fact, most networks tend to approximate the COPY network, retaining most of the information about the inputs. Learning dynamics in the rich regimes exhibit some early-stage compression for the coordinate transformation task, but the compressive code is quickly discarded in favor of improved posterior decodability. Notably, none of the networks exhibit the two-stage (fitting and compression) information plane trajectories observed in Schwartz-Ziv and Tishby (2017) despite being trained with stochastic gradient descent, with the findings of Saxe et al. (2019). Additioanlly, Bayesian transfer in the extrapolation training regime reveals that networks do not generalize well in all conditions – for the extrapolation condition, networks were not able to consistently surpass the calibration curve representing optimal decoding of the posterior mean with a fixed posterior uncertainty.

Figure 1B presents the results of our representational simplicity analysis. We expected that if a network's internal representation can transform a neural code to make relevant posteriors more easily decodable by downstream layers, then a simple linear posterior probe's performance should approach that of a high-capacity nonlinear probe over the course of learning (i.e., approaching the diagonal dashed line in Figure 1B). We find that this is not necessarily the case: though lazy-regime linear probes appear to linearize their code somewhat, rich-regime linear probes do not consistently improve their performance relative to the COPY network.

References

Beck, J. M., Latham, P. E., & Pouget, A. (2011). Marginalization in neural circuits with divisive normalization. *Journal of Neuroscience*, *31*, 15310-15319.

Ernst, M. O., & Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature*, *415*, 429-433.

Farrell, M., Recanatesi, S., & Shea-Brown, E. (2023). From lazy to rich to exclusive task representations in neural networks and neural codes. *Current Opinion in Neurobiology*, *83*, 102780.

Flesch, T., Juechems, K., Dumbalska, T., Saxe, A., & Summerfield, C. (2021). Rich and lazy learning of task representations in brains and neural networks. *arXiv*.

Haefner, R., Beck, J., Savin, C., Salmasi, M., & Pitkow, X. (2024). How does the brain compute with probabilities? *arXiv*.

Kording, K. P., & Wolpert, D. M. (2004). Bayesian integration in sensorimotor learning. *Nature*, *427*, 244-247.

Ma, W. J., Beck, J. M., Latham, P. E., & Pouget, A. (2006). Bayesian inference with probabilistic population codes. *Nature Neuroscience*, *9*, 1432-1438.

Orhan, E., & Ma, W. J. (2017). Efficient probabilistic inference in generic neural networks trained with non-probabilistic feedback. *Nature Communications*, *8*.

Pohl, S., Walker, E. Y., Barack, D. L., Lee, J., Denison, R. N., Block, N., ... Ma, W. J. (2024). Desiderata of evidence for representation in neuroscience. *arXiv*.

Saxe, A., Bansal, Y., Dapello, J., Advani, M., Kolchinsky, A., Tracey, B., & Cox, D. (2019). On the information bottleneck theory of deep learning*. *Journal of Statistical Mechanics: Theory and Experiment*, *2019*, 124020.

Schwartz-Ziv, R., & Tishby, N. (2017). Opening the black box of deep neural networks via information. *arXiv*.

Tishby, N., Pereira, F., & Bialek, W. (2000). The information bottleneck method. *arXiv*.

Walker, E. Y., Cotton, R. J., Ma, W. J., & Tolias, A. (2020). A neural basis of probabilistic computation in visual cortex. *Nature Neuroscience*, *23*, 122-129.

Walker, E. Y., Pohl, S., Denison, R. N., Barack, D. L., Lee, J., Block, N., ... Meyniel, F. (2023). Studying the neural representations of uncertainty. *Nature Neuroscience*, *26*, 1857-1867.