Bounded rationality in structured density estimation: Supplementary material

A Experimental details

A.1 Experiment 1

A.1.1 Participants

Experiment 1 recruited 21 participants (11 females, aged 18–25). All participants had provided informed consent before the experiment.

A.1.2 Cover story

Participants were told that they were apprentice magicians in a magical world. In this world, dangerous magic lava rocks were emitted from an unknown number of invisible volcano(es). On each trial, they observed past landing locations of lava rocks in a specific area (on the screen), and their job was to predict the probability density of future landing locations. More specifically, they were asked to draw a probability density by reporting, using click-and-drag mouse gestures, three key properties of the volcano(es), corresponding to the mean, the weight, and the standard deviation of a Gaussian component. They were told that their bonus payment depended on the accuracy of the reported predictive density.

A.1.3 Procedure and Design

On each trial, the landing positions of lava rocks were visualized as red dots and sequentially presented on a black line. The number of presented rocks (i.e., sample size) had two levels: 10 or 70. Each dot was presented for 166 ms, followed by a 166 ms empty screen. The landing positions were i.i.d. samples drawn from an unseen mixture distribution.

After observing all dots, participants needed to follow a two-stage procedure to report both (1) the predictive density of rocks' future landing positions, and (2) the underlying generative model. First, they marked the location of the volcano(es) by clicking on the black line. After marking all volcano(es), they pressed "F" to proceed to the next reporting stage. In the second stage, they chose a volcano by clicking the volcano icon and then moved the mouse to report the relative density of the landing position of future lava rocks emitted from this volcano. During the report, the overall density of the rock landing position was computed and presented in real-ime.

The true generative distribution set was composed of 24 distributions (Fig. 6A). The distributions were created by following a merge-from-four procedure. All distributions' overall standard deviations were about 5.3 cm. On each trial, a uniform jitter ([-5.3 cm, +5.3 cm]) was added to the mean of the true generative distribution.

Participants completed 2 (sample size levels) $\times$ 4 (true cluster number levels) $\times$ 6 (distribution subtype) $\times$ 3 (number of repeats) = 144 trials in total. The experiment length was about 105 minutes.

A.1.4 Generation of true distribution set

The set of true distributions was constructed following the steps below. First, we created 6 four-cluster Gaussian mixture distributions (bottom row in Fig. 6A) in which each component had the same SD of 0.72cm of visual angle and equal weight. The three center-to-center distances between their adjacent Gaussian components, denoted $[d_1, d_2, d_3]$ (from left to right, measured in cm), were chosen from the set of all permutations of $\{3.6, 4.65, 5.7\}$cm.

Second, each of the 6 four-cluster Gaussian mixtures went through "merging steps", inspired by the proposal step in the dynamic clustering algorithms (e.g. Reverse-jump MCMC) in statistics [36]. In each merging step, we chose two adjacent Gaussian components to merge into one, with the post-merger new component having the same zeroth, first and second

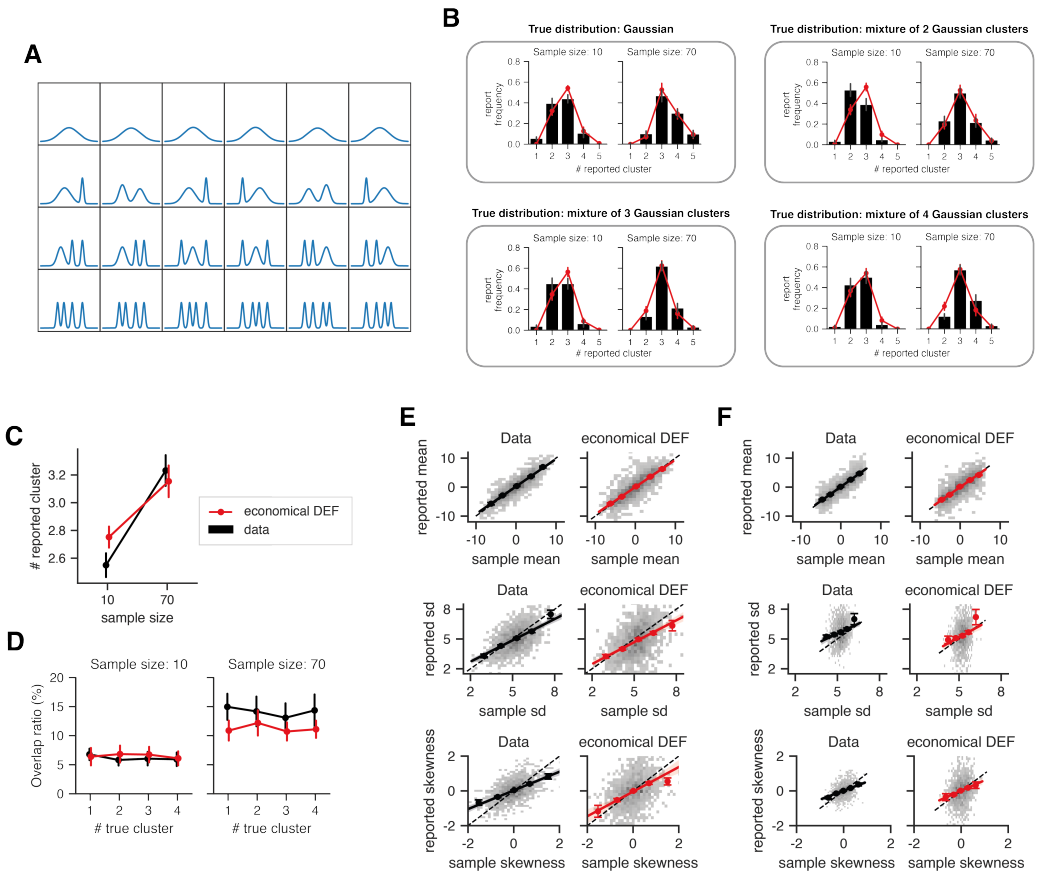


Figure 6: Detailed experimental design and results in Experiment 1. A, The distribution set. B–F shows the prediction of the fitted economical DEFs compared to participants' reports under different conditions. B, The relative frequency of the reported cluster number. C, The average reported cluster number. D, The overlap ratio between the reported clusters. E, The reported moments versus the sample moments in the 10 sample-size condition, with data (left sub-panels) contrasted with model prediction (right sub-panels). Three rows are for mean, sd, and skewness. In each panel, the grey heatmap denotes the distribution of responses of individual trials (collapsed across subjects). The 5 dots and error bars denote average response and standard error across subjects in 5 local data bins. The line denotes the regression line, with shading representing 95% confidence interval. F, The reported moments versus the sample moments in the 70 sample-size condition

moments as the combination of the two pre-merger components. The to-be-merged adjacent components were chosen in such a way that the post-merger Gaussian mixtures minimize KL-divergence $\mathrm{KL}(p_{\mathrm{pre}}||p_{\mathrm{post}})$. By applying the merging step iteratively, the original four-cluster Gaussian mixtures were transformed into three-cluster, two-cluster, and finally one-cluster mixtures.

A.1.5 Results

See Fig. 6 for the prediction of the fitted economical DEF.

A.2 Experiment 2

Experiment 2 recruited 21 participants (13 females, aged 18–25). All participants had provided informed consent before the experiment. One participant was excluded due to

Table 1: The distribution set in Experiment 2

Distribution	Paper	Skewed or Symmetric	Number of clusters	Number of modes
1	[24]	Symmetric	1	Unimodal
2	[24]	Symmetric	-*	Unimodal
3	[25]	Symmetric	2	Unimodal
4	[21]	Skewed	2	Unimodal
5	[21]	Skewed	2	Unimodal
6	[21]	Skewed	2	Unimodal
7	[25]	Skewed	2	Unimodal
8	[23]	Skewed	2	Bimodal
9	[25]	Symmetric	2	Bimodal
10	[22]	Symmetric	2	Bimodal
11	[22]	Symmetric	3	Trimodal
12	[20]	Skewed	3	Trimodal
13	[20]	Skewed	3	Trimodal
14	[20]	Skewed	3	Trimodal

* This is a uniform distribution with smoothed edges.

their task performance being an outlier (measured by the earth-mover distance between the reported predictive density and the true density, exceeding 3 standard deviations, z-score = -3.3).

Experiment 1 uses 14 representative Gaussian mixture distributions selected from previous studies (Table 1). The distribution set can be divided into four subtypes: unimodal-symmetric, unimodal-skewed, bimodal, and trimodal. Note that the number of modes is not necessarily equal to the number of latent Gaussian clusters. For example, a unimodal skewed distribution could be a mixture of two latent Gaussian clusters.

In each trial, participants observed 20 sequential samples drawn from one of the 14 distributions. The standard deviations of all distributions were rescaled to 4.8 cm. Each dot was presented for 166 ms, followed by a 166 ms empty screen interval. On each trial, a common uniform jitter ([-3.7 cm, +3.7 cm]) was added to the means of the clusters. To balance the skewness of the distribution in the experiment, we horizontally flipped the distribution in half of the trials. To explore participant's consistency in structure learning, we presented the same sample sequence for two times in seperate trials. Participants completed a total of 14 (distribution types) $\times$ 2 (flip or not) $\times$ 2 (number of random sequences) $\times$ 2 (number of repeats) = 112 trials. The experiment lasted about 90 minutes.

A.2.1 Results

See Fig. 7B–D&G for the prediction of the economical model.

Experiment 2 contained repeated trials with identical stimuli. Using these trials, we show in Fig. 7E that in repeated trials participants reported a different number of clusters just below 50% of the time. Similarly, our model could predict with an accuracy around 0.5, very close to the participants themselves. This shows that our model can predict human report close to the participants themselves.

To illustrate the robustness of the model fitting procedure, we run model recovery experiments. Given randomly chosen parameters for the full model, we generate 100 sets of synthetic stimuli, reset the parameters to new random values, and then fit the parameters on the synthetic dataset using the procedure described in the main paper. The results show that the recovered parameters are largely consistent with the random initial values (Fig. 7F, the average correlation between the source parameters and the fitted parameters is 0.84).

A.3 Experiment 3

Experiment 3 recruited 36 participants (21 females, aged 18–26). All participants had provided informed consent before the experiment.

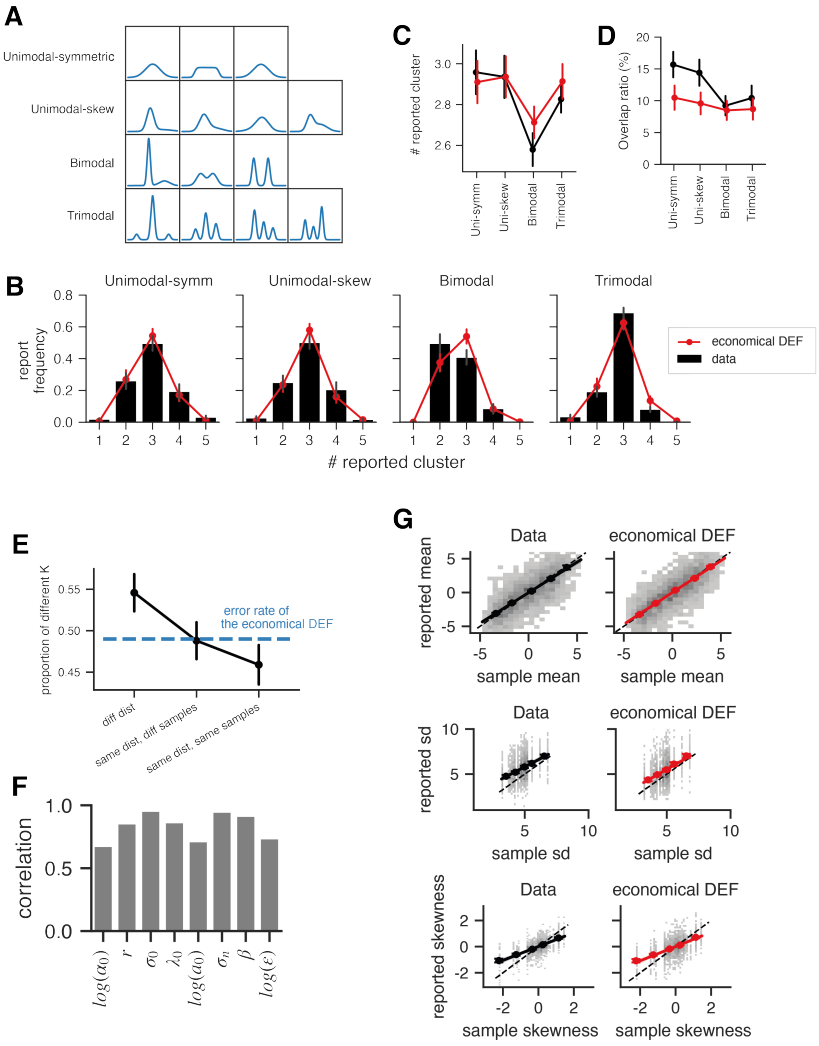


Figure 7: Detailed experimental design and results for Experiment 2. A, The distribution set. B-D&G shows the prediction of the fitted economical DEFs compared to participants' reports under different conditions. B, The relative frequency of the reported cluster number. C, The average reported cluster number. D, The overlap ratio between the reported clusters. E, The proportion that the reported numbers of clusters are different when two trials have (1) different distributions, (2) same distribution but different samples, or (3) same distribution and samples. F, The quality of model recovery, measured by the correlations between the source parameters and the fitted parameters. G, The reported moments versus the sample moments.

In Experiment 3, participants needed to learn the distribution of numeric values and report their belief of the distribution by entering numbers on the keyboard. On each trial, the horizontal coordinates of lava rocks were shown one-by-one on the screen, with each coordinate presenting for 1.5 seconds, followed by a 0.5-second empty screen. After observing all coordinates, participants were required to first report the number of volcanoes. Then, they were required to enter the location and the relative eruption frequency of each volcano.

The true distribution set was composed of 2 unimodal distributions: one is a Gaussian distribution, and the other is a skewed Gaussian mixture with 2 wide clusters (Fig. 2A). Participants completed 3 (distribution type) $\times$ 8 (number of repeats) $=$ 24 trials in total.

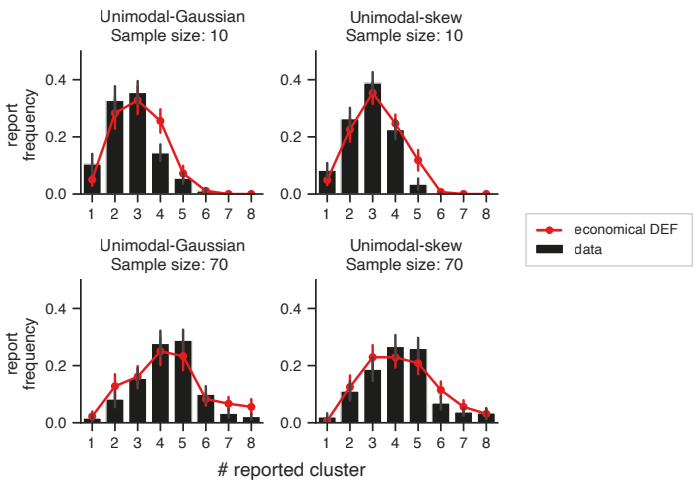


Figure 8: The relative frequency of the reported cluster number in Experiment 3.

A.3.1 Results

See Fig. 8 for the prediction of the economical model.

B Density estimation framework details

B.1 Rational component

The rational component should take the sequential observations $\mathbf{x}_T$ and predict the internal construct properties $\boldsymbol{\varphi}^r$ reported by a participant. From a computational perspective, any simulation-based model could be used for this component, as long as it can produce an internal construct based on $\mathbf{x}_T$. In this work, we adopt the rationale of Bayesian inference given a plausible generative process that could have produced the observations $\mathbf{x}_T$, with approximations mandated by cognitive constraints. Since the behavioral report is derived from a subjective belief of the unobserved distribution in the environment, the generative process in the participant's mind describes a subjective prior (or a construct) over distributions. As such, we call this generative process an "internal construct prior" (ICP).

The ICP in our main model is inspired by the nonparametric Gaussian mixture model. A popular instantiation of the said model is defined through a CRP prior over the cluster assignments, and a conjugate prior over the distribution of each cluster, abbreviated as CRP-GMM. It is appealing for our purpose because the implied generative process of $\mathbf{x}_T$ is sequential, similar to how participants observe the $\mathbf{x}_T$. To make this prior more flexible, we extend the cluster assignment prior by introducing additional parameters in (1) to control the expansion decay rate (r) and count distortion rate β. The effects of these two parameters on the prior distribution of partitions are shown in Fig. 9. While the number of clusters in the ordinary CRP grows, the decaying parameter α in our extension substantially slows down the introduction of new clusters as the model size increases. As of the distortion rate β, when $\beta < 1$, there is weaker rich-get-richer effect, giving more evenly distributed cluster sizes and higher entropy in the cluster assignment distribution, while a $\beta > 1$ produces more extreme cluster sizes and lower entropy of the cluster assignment distribution.

B.1.1 Inexchangeability of the the economical ICP

The exchangeability of a cluster assignment prior is often desirable for online data modeling, because it ensures that the posterior of cluster assignment is invariant to the order (permutation) of data. However, in this work, we are interested in a prior that governs how humans construct a density model online. Our discussion on cognitive constraints in

Section 1 suggests that achieving order invariance requires implausible computations, so we propose to relax order invariance for the purpose of modeling human cognition.

Here we show by counterexamples the proposed ICP cluster assignment prior (1) is not exchangeable. First, for the case $\beta = 1$ and $r \neq 0$. consider the partition $\{\{z_1, z_2\}, \{z_3\}\}$. We have

$$\begin{aligned}\mathbb{P}(z_1 = 1, z_2 = 1, z_3 = 2) &= \mathbb{P}(z_1 = 1)\mathbb{P}(z_2 = 1|z_1 = 1)\mathbb{P}(z_3 = 2|z_1 = 1, z_2 = 1) \\ &= 1\frac{1}{1 + \alpha_0 e^{-r}}\frac{\alpha_0 e^{-r}}{2 + \alpha_0 e^{-r}},\end{aligned}$$

which is not the same as the probability of a permuted but equivalent partition,

$$\begin{aligned}\mathbb{P}(z_3 = 1, z_1 = 2, z_2 = 2) &= \mathbb{P}(z_3 = 1)\mathbb{P}(z_1 = 2|z_3 = 1)\mathbb{P}(z_2 = 2|z_3 = 1, z_2 = 2) \\ &= 1\frac{\alpha_0}{1 + \alpha_0}\frac{1}{2 + \alpha_0 e^{-r}}.\end{aligned}$$

Then, for the case $\beta \neq 1$ but $r = 0$, consider the partition $\{\{z_1, z_3, z_5\}, \{z_2, z_4\}\}$. We have

$$\mathbb{P}(z_1 = 1, z_2 = 2, z_3 = 1, z_4 = 1) = 1\frac{\alpha_0}{1 + \alpha_0}\frac{2 \cdot \frac{1^\beta}{1^\beta + 1^\beta}}{2 + \alpha_0}\frac{3 \cdot \frac{2^\beta}{2^\beta + 1^\beta}}{3 + \alpha_0} = \frac{3\alpha_0 2^\beta}{(2^\beta + 1)\prod_{t=1}^{3}(t + \alpha_0)},$$

which is not the same as the probability of a permuted but equivalent partition,

$$\mathbb{P}(z_1 = 1, z_3 = 1, z_4 = 1, z_2 = 2) = 1\frac{1}{1 + \alpha_0}\frac{2}{2 + \alpha_0}\frac{\alpha_0}{3 + \alpha_0} = \frac{2\alpha_0}{\prod_{t=1}^{3}(t + \alpha_0)}.$$

Therefore, the prior defined by (1) is not exchangeable in general if $\beta \neq 1$ or $r \neq 0$.

B.1.2 Sufficient statistics

For completeness, we give the explicit expressions of the sufficient statistics in (3) as

$$n^i_{t,k} = \sum_{\tau=1}^{t} \mathbb{1}[z^i_\tau = k],\ \bar{\mu}^i_{t,k} = \frac{1}{n^i_{t,k}}\sum_{\tau=1}^{t} \mathbb{1}[z^i_t = k]x_t,\ \bar{\sigma}^i_{t,k} = \frac{1}{n^i_{t,k}}\sum_{\tau=1}^{t} \mathbb{1}[z^i_t = k](x_t - \bar{\mu}^i_{t,k})^2. \quad (8)$$

We assume that participants maintain these sufficient statistics when building their internal constructs, and also report them at the end of the trial, after normalizing $\mathbf{n}^i$ to obtain the weights. Note that, unconventionally, we denote the *variance* by σ.

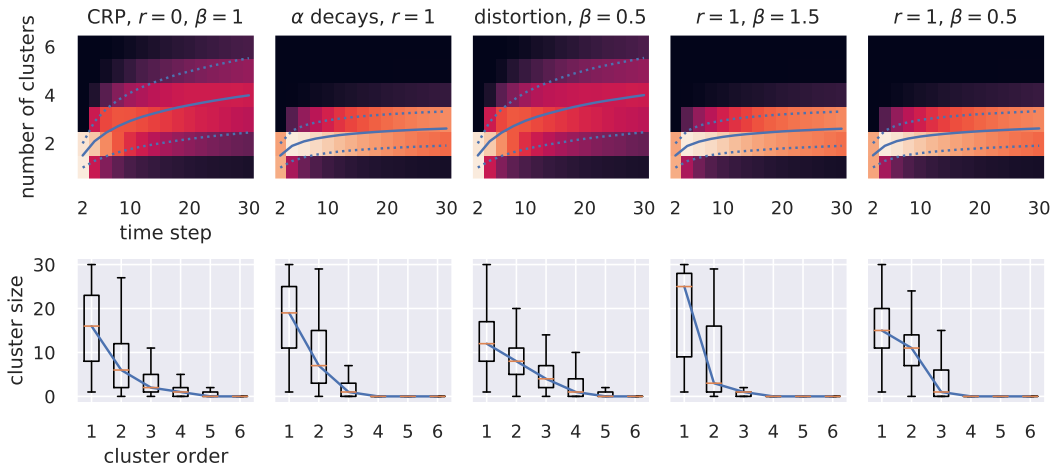


Figure 9: Effects of the decay rate and distortion rate β in (1) on the distribution of partitions. Heatmaps on the top row show distributions of the number of clusters as a function of time steps. Brighter color means higher probability. Blue solid line indicates the mean, blue dotted lines indicate 1 standard deviation. The bottom row shows the distribution of cluster sizes at $t = 30$. Yellow bar indicates median.

B.1.3 Baseline batch ICP

As both the economical ICP in Section 3.1 and the proposed fitting algorithm (to appear in Appendix B.3) are new, it is important to compare the economical ICP and the exchangeable ICP, CRP-GMM, with another baseline ICP, all fit by the same algorithm. If we expect the baseline ICP to provide a worse description of human cognition than CRP-GMM on this task, then the proposed algorithm must be able to produce a less favorable model comparison result for this baseline after fitting them to human data. We choose a batch ICP as the baseline: it generates all observations as i.i.d. samples conditioned on cluster assignments, but, unlike the CRP, the cluster assignment prior is time-invariant and is thus exchangeable; inference over this ICP produces order-invariant posteriors. Because our task has a strong sequential nature, we expect this batch ICP to be a worse descriptor of human cognition for this task.

Specifically, this batch ICP p_{B} maintains a truncated Poisson prior distribution over $K_{\max}$ submodels of mixture distributions.

$$p_{\mathrm{B}}(K) \propto \begin{cases} \mathrm{Poisson}(K; \bar{K}), & 1 \le k \le K_{\max}; \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

The K'th submodel is a Gaussian mixture of K components. Each submodel has a symmetric Dirichlet prior over the cluster weights, and each component has mean and variance following the conjugate Gauassian-Inverse-χ^2 distribution, as in (2). The truncated Poisson prior is a heuristic choice; we also tried other priors, such as a Categorical distribution supported on $K = 2$ and $K = 3$, or a truncated Poisson with mean dependent on the sample size (10 vs 70). These choices improved the AIC only insignificantly.

Inference in each submodel The following description applies to each of the $K_{\max}$ submodels. Given the observed noisy data $\tilde{\mathbf{x}}_T^i$, each submodel produces an inferred cluster property φ_K^i, which are the sufficient statistics of the cluster property posteriors returned by the variational-EM algorithm (see Chapter 10.2 of [37]). We now drop the superscript i to reduce clutter, bearing in mind that all the variables are samples depending on a noisy observation $\tilde{\mathbf{x}}_T^i$ and a sampled K^i. This algorithm alternates between the E-step: updating a Dirichlet posterior over the cluster weights; and the M-step updating a Gaussian-Inverse-χ^2 posterior over the cluster means and variances. It turns out that the posterior over the cluster parameters φ_K depends on a set of sufficient statistics similar to (8), except that the cluster assignments are now soft as computed by the responsibilities $r_{t,k}$ for each x_t and $k \in \{1, \dots, K\}$ during the variational E-step.

After the variational-EM procedure converges, we perform a hard cluster assignment for each observation x_t by choosing the cluster with the highest responsibility,

$$z_t = \operatorname*{arg\,max}_{k \in \{1, \dots, K_{\max}\}} r_{t,k}.$$

This ensures that the cluster weights are quantized, as is the case for the particle-based inference methods on the other two ICPs—we do not want to introduce additional effects to the variational model due to having continuous support for cluster weights. Thus, at the last time step $t = T$, the cluster weights are determined by $n_k = \sum_{t=1}^{T} \mathbb{1}[z_t = k]$. The cluster mean μ_k and variance σ_k still depend on the original unquantized responsibilities and per-cluster sufficient statistics similar to (8) but with indicators $\mathbb{1}[z_t = k]$ replaced by responsibilities $r_{t,k}$. As a result, for each submodel with K clusters, the inferred cluster properties are summarized by $\varphi_K := [n_k, \bar{\mu}_k, \bar{\sigma}_k]_{k=1}^K$.

Submodel selection After obtaining sufficient statistics for all K models, the batch rational component selects the submodel with the largest marginal likelihood given the hard cluster assignments, computed through Student's t marginal densities as in (4), minus a penalty of a weighted model size

$$K_{\mathrm{B}} := \operatorname*{arg\,max}_{K \in \{1, \dots, K_{\max}\}} \{p_{\mathrm{B}}(\mathbf{x}_T | \hat{\varphi}_K) - \gamma K\} \tag{10}$$

where γ is a trainable parameter. This penalty is consistent with how AIC penalizes model complexity. Then, the submodel with K_{B} clusters is selected, passing $\varphi_{K_{\mathrm{B}}}$ to the aleatoric component. As such, the batch-based rational component differs from the CRP-GMM in the following ways:

1. In the batch rational component, the cluster assignments are performed by variational inference rather than a particle filter;
2. the batch rational component performs model selection that penalizes large models after inferring multiple submodels, whereas CRP-GMM embeds the preference for smaller models in the cluster assignment prior.

Likelihood approximation These inferred cluster properties are passed to the aleatoric component, giving (slacked) number of cluster $\hat{K}_{\mathrm{B}}$ and the predicted $\hat{\varphi}_{\mathrm{B}}$. This amounts to a single simulation of the batch DEF. To compute the likelihood, the batch DEF still needs to marginalize out the visual noise and the slacked $\hat{K}_B$ in the aleatoric component. To this end, we run a large number of simulations, each with an independent draw of noisy observations $\tilde{\mathbf{x}}_T^i$, giving the predicted cluster properties $\hat{\varphi}_{\mathrm{B}}^i$. The likelihood $p(\varphi^r|\mathbf{x}_T)$ for the batch model is then approximated by (7).

The batch rational model and the aleatoric component combine to give the batch DEF, which is used to benchmark the economical ICP and exchangeable ICP in the corresponding DEFs.

B.2 Aleatoric component

Here, we explain in more detail the aleatoric component described in Fig. 3. As discussed in Section 3.2, one challenge in modeling this dataset is that the dimensionality of the internal construct varies across trials. This requires that the model be able to produce variable-dimensional predictions. In order to fit the DEFs by maximum-likelihood, the DEF must place nonzero probability to all possible numbers of clusters. We thus defined the distribution $p_{\mathrm{A}}(\hat{K}|K)$ in (5) supported on $\{1, \ldots, K_{\max}\}$, and restrict the maximum number of clusters to $K_{\max}$ to be the largest number of clusters ever reported by participants in an Experiment. One can also define other slack distributions over K with decaying tails to avoid an explicit upper bound.

If there is no slack, then the predicted cluster properties are as inferred. If the participant slacks with probability ϵ and commits to a prediction $\hat{K}^i$ that does not agree with the inferred K^i from the rational component, they must modify the inferred φ^i so that there are $\hat{K}^i$ clusters. We assume that, during modification, the participant should keep the overall distribution roughly intact. We propose the following deterministic procedure, denoted by $f(\varphi^i, \hat{K}^i)$, which recursively increases or decreases the number of clusters in φ^i until there are $\hat{K}^i$ left.

Removing the smallest cluster. This happens whenever φ^i has more clusters than $\hat{K}^i$. We simply remove the cluster with the smallest weight. An alternative is the following merging strategy: take the cluster with the smallest weight, and merge into its nearest neighbor. The merged distribution has weight equal to the sum of the weights of the clusters merged, and has mean and variances equal to the effective mean and variance of the two. More precisely, for two clusters with properties $[w_1, \bar{\mu}_1, \bar{\sigma}_1]$ and $[w_2, \bar{\mu}_2, \bar{\sigma}_2]$ (note that we denote the variance by σ), the merged cluster has properties

$$[w_1 + w_2, \mu_m, \sigma_m],$$

where $\mu_m = w_1\bar{\mu}_1 + w_2\bar{\mu}_2$ and $\sigma_m = w_1(\bar{\sigma}_1 + \bar{\mu}_1^2) + w_2(\bar{\sigma}_2 + \bar{\mu}_2^2) - \mu_m^2$. Our results show that the removal strategy produced better AIC than the merging strategy.

Splitting the largest cluster. This happens whenever φ^i has fewer clusters than $\hat{K}^i$. We take the cluster with the largest weight and split it into two clusters. The new clusters are centered at equal distance from and on two sides of the original cluster, and their variance is a scaled version of the variance of the original cluster. Specifically, denote the properties

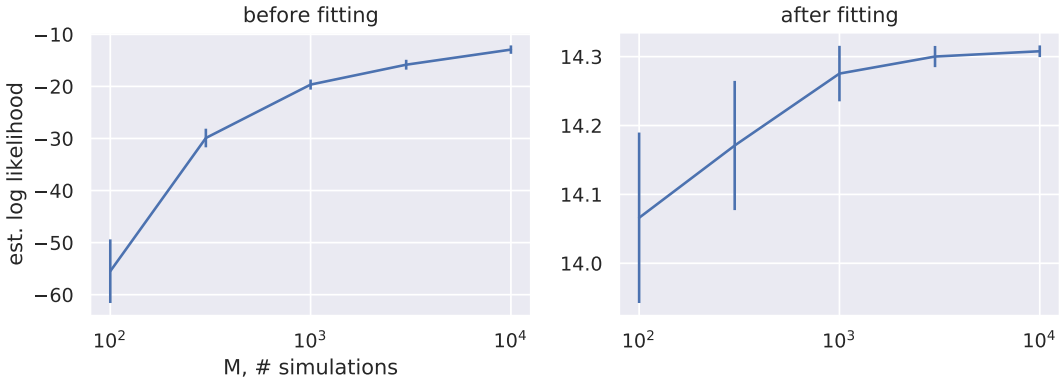


Figure 10: Estimated log-likelihood of data for one participant given different numbers of simulations. We show the mean and standard deviations of the estimate based on 30 runs for each value of M.

of this cluster to be split by $[w, m, \sigma]$, we set the cluster properties of the new clusters to be

$$\left[\frac{w}{2},\ m - b_\sigma\sqrt{\sigma},\ \ s_\sigma\sigma\right], \quad \left[\frac{w}{2},\ m + b_\sigma\sqrt{\sigma},\ \ s_\sigma\sigma\right]. \tag{11}$$

where $b_\sigma > 0$ and $0 \leq s_\sigma \leq 1$ are free parameters to be optimised. We also tested a version of this strategy where these two parameters are fixed at $b_\sigma = \frac{\sqrt{3}}{2}$ and $s_\sigma = 1/4$, but this gave worse AIC.

Noise processes. The noise processes defined in (6) are given by

$$p_w(\mathbf{w}; \hat{\mathbf{n}}) = \text{Dirichlet}(\mathbf{w}; c(\hat{\mathbf{n}} + 1)), \tag{12}$$

$$p_\mu(\boldsymbol{\mu}^r; \hat{\boldsymbol{\mu}}) = \prod_{k=1}^{\hat{K}} \mathcal{N}(\mu_k^r; \hat{\mu}_k, \sigma_\mu), \tag{13}$$

$$p_\mu(\boldsymbol{\sigma}^r; \hat{\boldsymbol{\sigma}}) = \prod_{k=1}^{\hat{K}} \log\mathcal{N}(\sigma_k^r; \hat{\sigma}_k, \sigma_v). \tag{14}$$

where $\hat{K}$ is implied from $\hat{\varphi}$. The addition by 1 in (12) ensures that the mode of the Dirichlet distribution is equal to $w \propto \hat{n}$. The scaling parameter c changes the confidence. The variances σ_μ and σ_ν are free parameters.

B.3 Fitting algorithm

The MC estimator for the likelihood in (7) is unbiased. However, the corresponding log-likelihood estimator, adopted in practice for numerical stability, is only consistent and produces a nonzero bias for a finite number of simulations M. This is due to Jensen's inequality. Suppose each simulation provides a log-likelihood estimate of ℓ_i conditioned on some latent variables (such as $\mathbf{z}_T^i$ and $\hat{K}^i$). Let the true conditional likelihood be X so that $e_i^\ell \sim X$. then the estimated marginal log-likelihood is

$$\mathbb{E}\left[\log\left(\frac{1}{M}\sum_{i=1}^{M} e^{\ell_i}\right)\right] \leq \log\left(\mathbb{E}\left[\frac{1}{M}\sum_{i=1}^{M} e^{\ell_i}\right]\right) = \log(\mathbb{E}[X]) \tag{15}$$

Note that $\mathbb{E}[X]$ is the marginal likelihood. Fortunately, this bias is downwards, meaning that the expected MC estimate provides a lower bound on the true log-likelihood Still, we must use a large M during training and evaluation, as this reduces the variance of the empirical average in (15) and thus lowers the bias. We show in Fig. 10 the dependence of the estimated log-likelihood for different numbers of simulations, using data from a randomly

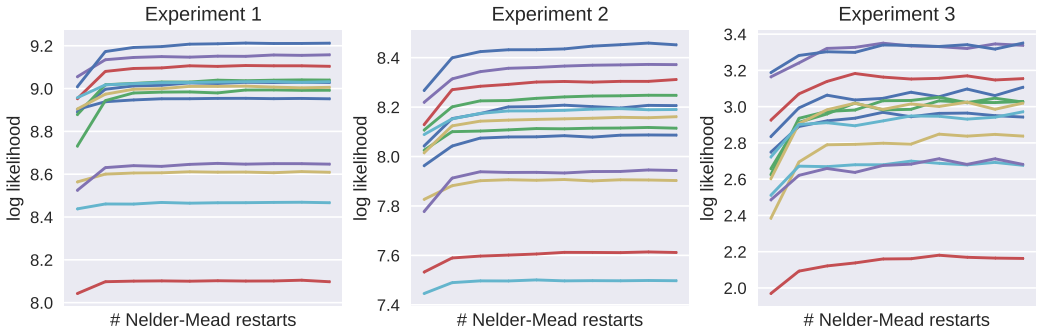


Figure 11: Log-likelihood of different DEFs in ablation studies, including the economical and CRP-GMM DEFs.

picked participant in Experiment 3. When the model is untrained (left panel), the bias does not completely go away even at $M = 10\,000$, but the variance is much smaller than using fewer M's. When the model is well-trained, both the bias and variance of the estimated log-likelihood are substantially reduced (right panel). Thus, the likelihood estimates become more reliable as the DEF fits the data better.

Thanks to the small estimation variance, we are able to fit the parameters of the DEFs using a gradient-free optimization routine Nelder-Mead implemented in the NLOpt package [2]. During training, we estimate the log-likelihood using $M = 10\,000$ parallel simulations on NVIDIA GTX 1080 and A100 GPUs. The latter GPU provides a likelihood estimate within 3 seconds for Experiment 2, and 5 seconds for Experiments 1 and 3. The Nelder-Mead routine typically converges to a 0.001 relative precision on parameters within 300 iterations.

We restart Nelder-Mead 10 times with parameters found from the previous optimization, as this avoids early convergence of Nelder-Mead. To further avoid local optima, we repeat this whole procedure (with 10 restarts) 10 times with different random seeds. We can then check if our algorithm has found a good solution by taking the maximum likelihood found across the 10 repeats. If this solution is reliable, then we should observe that this maximum is stable across the 10 restarts. We then take the best solution among the 10 random seeds at the last repeat as the fitted DEF parameters.

Fig. 11 shows the best (among seeds) log-likelihood across 10 restarts, averaged over all participants, for each DEF setup (see Appendix B.5.1). During the first repeat, the log-likelihood increases and stabilizes. Because we do not keep the best log-likelihood across the restarts, we see small fluctuations across the restarts. We see a very slow increase of log-likelihood for some DEFs in Experiment 3, but the relative rank of different DEFs is mostly preserved.

B.3.1 Fitting the batch DEF

For the Batch DEF, each simulation maintains $K_{\max}$ submodels before the comparison in (10). Because the cluster weights posterior can be computed during variational inference, unlike in the CRP where we used particles, we do not need as many simulations, and so we run $M = 100$ simulations of each of the $K_{\max}$ submodels. The output of the Batch rational component is then fed into the same aleatoric component for a fair comparison between the ICPs.

B.4 Particles versus MC simulations

There is an important distinction between the number of particles and the number of MC simulations. In this work, we assume that the participant uses a single particle for inferring the cluster assignments, but this is not observed from the experimenter's viewpoint. The reported internal construct may be associated with one out of many possible cluster

[2] https://github.com/stevengj/nlopt

Table 2: Table of all model parameters.

Parameter name	Symbol	DEF component	Equation	Log-space?	In default?	Notes
Expansion rate	α_0	rational	(1)	Y	Y	
Decay rate	r	rational	(1)	N	N	optimized in economical, fixed to 0 in exchangeable
Cluster count distortion	β	rational	(1)	Y	N	optimized in economical, fixed to 1 in exchangeable
Prior mean	μ_0	rational	(2)	-	-	fixed to zero, not fitted
Pseudocount of prior mean	λ_0	rational	(2)	Y	Y	
Prior variance	σ_0	rational	(2)	Y	Y	
Pseudocount of prior variance	a_0	aleatoric	(2)	Y	Y	
Slack on cluster count	ϵ	aleatoric	(5)	Y	Y	
Concentration scaling in reporting cluster weight	c	aleatoric	(6), (12)	Y	Y	
Gaussian noise variance in reporting cluster mean	σ_v	aleatoric	(6), (13)	Y	Y	
Gaussian noise variance in reporting cluster log variance	σ_m	aleatoric	(6), (14)	Y	Y	
Visual noise variance	σ_n	-	(7)	Y	Y	
Poisson prior mean	$\overline{K}$	batch rational	(9)	Y	N	only in the baseline batch ICP
Weight on model size penalty	γ	batch rational	(10)	Y	N	only in the baseline batch ICP
Cluster weight prior	-	batch rational	-	Y	N	only in the baseline batch ICP

assignments. As such, we run many simulations of the single-particle particle filter. The number of MC simulations is M, and each simulation is indexed by i. Since we stick to the single-particle assumption throughout the paper, we do not need an additional index for the number of particles within each MC simulation.

Future work may consider multiple particles. In this case, each MC simulation will contain multiple sequences of cluster assignments. Maintaining multiple cluster assignment sequences allows re-assignment or re-weighting of the observations at each time step, achieving some retrospective correction. However, it is unknown how the participant makes a decision on reporting the cluster properties from multiple samples, especially when the number of clusters do not agree across different particles. We thus leave multiple-particle models for future work.

B.5 Additional modeling results

After training the DEFs, we evaluate the log-likelihood using a large number of simulations: 10^6 for the sequential DEFs in our main results, and 10^5 for the baseline batch DEF.

B.5.1 Ablation studies on DEF

Our initial experiment design used the exchangeable DEF (with a CRP-GMM rational component) as a reference model. We explored the model space by making small modifications to this DEF motivated by resource constraints and other heuristics. Here, we show that the two modifications in the economical DEF, namely the **Decay** in expansion rate α_t and **Distort**ion in cluster count $n_{t,k}$ produce reliable improvements over the reference exchangeable DEF. The DEFs resulting from other modifications either did not produce reliable improvements or were insignificant compared to the reference model. All model parameters are listed in Table 2. We introduce the modifications below.

Decay: adding a model-size dependent decay rate r, as in (1).

Distort: adding an exponential transformation to the size of the clusters, as in (1).

Fixed Splitting: when $\hat{K}^i < K^i$ and splitting a cluster, the parameters of the splitting are fixed with $b_\sigma = \sqrt{3}/2$ and $s_\sigma = 1/4$

Merge Cluster: when $\hat{K}^i < K^i$, instead of removing a cluster, the function $f(\varphi, \hat{K})$ merge the smallest cluster into the cluster with closest mean. The cluster properties of the new component is computed by moment matching, as detailed in Appendix B.2.

Local MAP: instead of sampling the cluster assignment according to (4), take the cluster with largest posterior probability. This is the local MAP procedure described in [15].

Constant Variance: instead of updating the variance of each cluster according to (3), the cluster mean is fixed to the trainable parameter σ_0

Fixed Mean: instead of updating the mean of each cluster according to (3), the mean is fixed at the first observation assigned to the cluster. This checks if participants are able to update the mean or simply remember the first observations assigned to the clusters.

Fixed Mean Confidence: instead of fitting λ_0 as a parameter, we fix it at $\lambda_0 = 0.01$

No Visual Noise: do not add Gaussian noise to the observations.

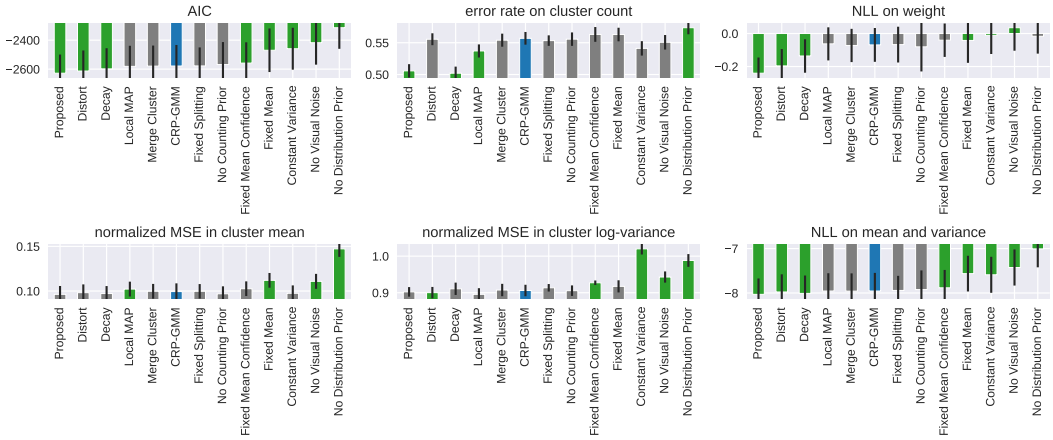
No Distribution Prior: when computing the per-step assignment posterior in (4), the likelihood is a Student's t-distribution obtained from having a conjugate prior over the Gaussian distribution parameters. Instead of computing this likelihood using the t-distribution, this modification computes this likelihood by a Gaussian with mean $\mu_{t,k}$ and variance $\sigma_{t,k}$.

No Counting Prior: in (1), instead of using the $n_{t,k}$ maintained for each cluster, use the average cluster size.

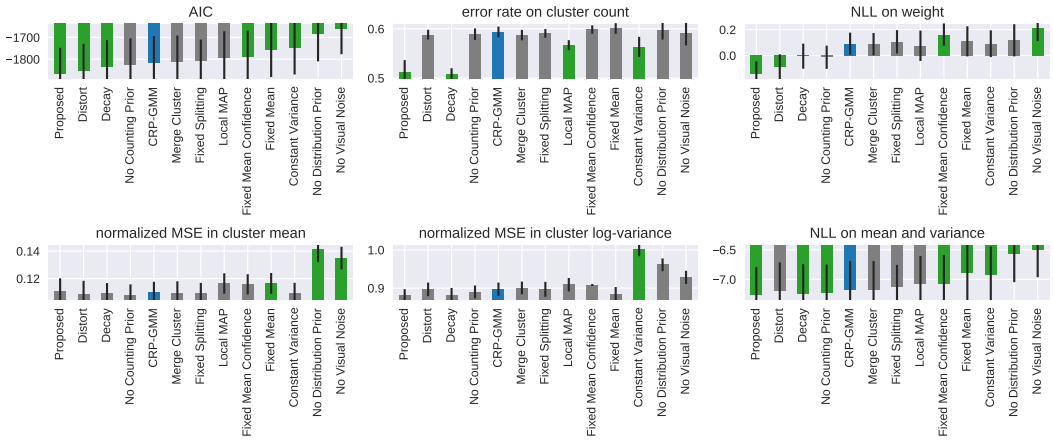
The results of these DEFs, averaged over participants, are shown in **??**. Clearly, only Decay and Distort produced reliable improvement on AICs compared to the CRP-GMM DEF across all Experiments. All other modifications that deviate from a rational approximation

of the Bayes rule (e.g. Fixed Mean, No Counting Prior, etc.) resulted in significantly worse fit to the reported internal constructs from our participants. We also see that removing visual noise is detrimental to the quality of the fit.

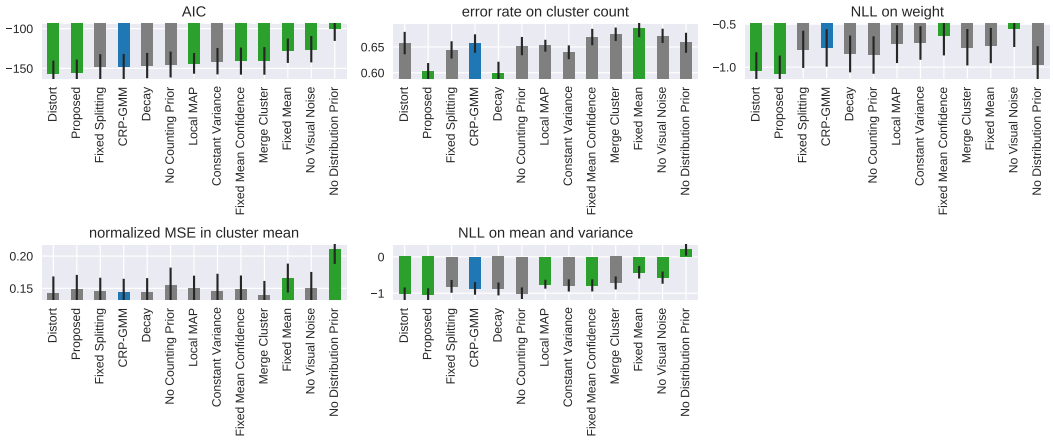
The stability of the likelihood approximated from (7) is shown in Fig. 11. All DEFs fitting converged well as the number of restarts increases, except for a few worse DEFs in Experiment 3.



(a) Experiment 1



(b) Experiment 2



(c) Experiment 3

Figure 12: Model ablation comparisons for all Experiments. Lower values are better. Error bars are 1 sems. Blue bar indicates the reference DEF with CRP-GMM as the rational component. Green bars indicate significant differences to the reference model (Wilcoxon signed-rank test, p<0.05), and grey bars indicate insignificant comparison. The Proposed (Distort + Decay) is the economical ICP.

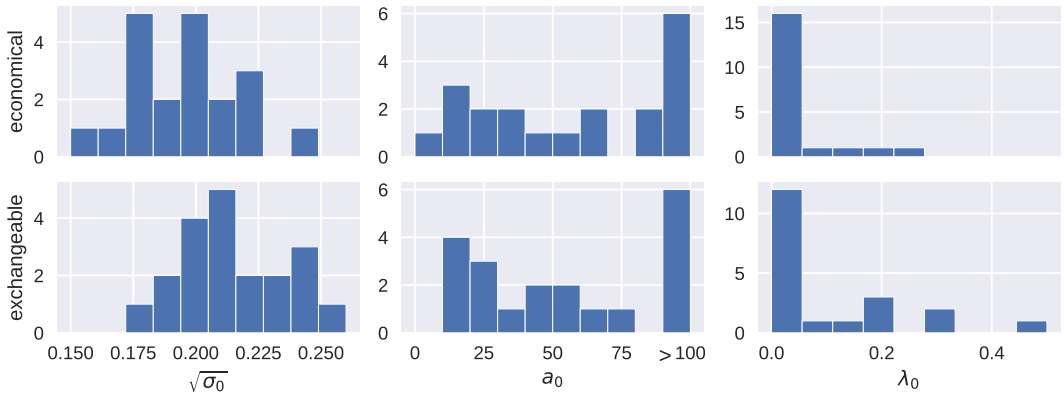


Figure 13: Distribution of fitted ICP parameters using data in Experiment 2.

B.5.2 Strong prior on cluster width

We found in most fitted ICPs that the prior cluster standard deviation is small (see Fig. 5 and Fig. 13), and the confidence indicated by its pseudocount a_0 is much greater than zero. This high confidence contrasts with the low pseudocount associated with the prior cluster mean λ_0. This means that participants could learn the cluster mean mostly driven by the observations while failing to adapt to the cluster uncertainty. This contributes to the large number of clusters seen when there is a single Gaussian in the true data distribution. We noted that Gershman and Niv [35] also used a relatively high $a_0 = 10$ parameter for their task.

References

[36] Sylvia Richardson and Peter J. Green. On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 1997.

[37] Christopher M. Bishop. *Pattern Recognition and Machine Learning*, Springer, 2006.