RewCon Modeling Summary

The rational model of categorization (RMC; Anderson, 1991) was fitted to participants' responses during the training phase. In particular, we used the particle filter version of the RMC (Sanborn, Griffiths, & Navarro, 2010). The modeling exercise is summarized in this document. First, I briefly describe the RMC, with the aim to introduce notations I use to describe model-based estimates.

1 Model — Rational model of categorization (RMC)

Suppose a learner has observed $n-1$ objects $\{x_1, x_2, \ldots, x_{n-1}\}$ with corresponding category labels $\{y_1, y_2, \ldots, y_{n-1}\}$. In the RewCon experiment, x_i is a pair of cues presented in the ith trial, and y_i is a corresponding category (hat or glove). Exemplar (e.g., green hat or black glove) is denoted as y'_i. In the RMC, each of these objects fits into a mental cluster. The cluster label for the ith object is denoted as z_i.

1.1 Drawing an inference with the RMC

Then, the probability that the nth object fits into category w is expressed as follows:

$$\begin{aligned} p(y_n = w \mid x_n) &= \sum_{k \in \mathbb{Z}} p(z_n = k \mid x_n)\, p(y_n = w \mid z_n = k) \\ &= \sum_{k \in \mathbb{Z}} \frac{p(z_n = k)\, p(x_n \mid z_n = k)}{p(x_n)}\, p(y_n = w \mid z_n = k) \\ &= \sum_{k \in \mathbb{Z}} \frac{p(z_n = k)\, p(x_n \mid z_n = k)}{\sum_{s \in \mathbb{Z}} p(z_n = s)\, p(x_n \mid z_n = s)}\, p(y_n = w \mid z_n = k). \end{aligned} \tag{1}$$

Here, $\mathbb{Z}$ is a set of all the possible clusters to which the nth object can be assigned. The three terms in Equation 1 are described below in turn.

First, the probability that the nth object fits into a cluster k is given by:

$$p(z_n = k) = \begin{cases} \dfrac{c\, m_k}{(1-c) + c\,(n-1)} & \text{if} \quad m_k > 0 \\[2ex] \dfrac{(1-c)}{(1-c) + c\,(n-1)} & \text{if} \quad m_k = 0 \end{cases} \tag{2}$$

where c is a parameter called the coupling probability and m_k is the number of objects assigned into cluster k.

Following Anderson (1991) and Sanborn et al. (2010), we assume that an object has independent dimensions. Therefore,

$$p(x_n \mid z_n = k) = \prod_{d \in D} p(x_{n,d} \mid z_n = k), \tag{3}$$

Questions concerning this document should be directed to Takao Noguchi (t.noguchi@ucl.ac.uk).
Last updated: August 31, 2015.

where D is a set of dimensions in which an object is described (i.e., first and second cues). The above term is computed with

$$p(x_{n,d} = v \mid z = k) = \frac{B_{v,d} + \beta_c}{m_k + J_d\, \beta_c}, \tag{4}$$

where $B_{v,d}$ is the number of objects in cluster k with value of v on dimension d, and J_d is the number of values which an object can take on dimension d (i.e., four in the RewCon).

Similarly, the probability that the nth object has category label w, given a cluster, is given by:

$$p(y_n = w \mid z = k) = \frac{B_w + \beta_l}{B_. + J\, \beta_l}, \tag{5}$$

where B_w is the number of observed objects with category label w in cluster k, $B_.$ is the number of object in cluster k, and J is the number of category labels.

1.2 Learning with the RMC

The learning in the RMC is to assign an object into a cluster. The cluster assignment at the nth trial, for example, is performed with the probability of each cluster. The probability that an object fits into cluster k is computed as

$$p(z_n = k \mid x_n, y_n, y_n') \propto p(z_n = k)\; p(x_n \mid z_n = k)\; p(y_n \mid z_n = k)\; p(y_n' \mid z_n = k). \tag{6}$$

This is computed with Equations 2, 3 and 5.

1.3 Parameter Estimation

The RMC, when applied to the RewCon experiment, has four parameters: coupling probability c, sensitivity parameter for cues β_c, sensitivity parameter for category label β_l, and sensitivity parameter for exemplar β_e. Following Sanborn et al. (2010), the number of particle is set to 1.

We estimated the most likely parameter values given participants' behavioral responses. The RMC with the particle filter, however, is stochastic and the model prediction can vary from simulation to simulation, even with the same set of parameter values. Thus for each iteration in parameter estimation process, we simulated the model 2,000 times to obtain the prediction and its variance.

Also to avoid over-fitting, we assumed that participants have varying values for the coupling probability c but share values for the sensitivity parameters (β_c, β_l, and β_e). Also, the sensitivity parameters for the category labels and exemplars are assumed identical (i.e., $\beta_l = \beta_e$). Therefore, the estimated parameters are: c for each participant, and β_c and β_l ($= \beta_e$) shared across participants.

With the prior distribution $\mathcal{U}(0, 1)$ for the coupling probability and $\mathcal{U}(0.01, 10)$ for the sensitivity parameters, we obtained maximum a posterior (MAP) estimation with the metric optimization engine (`http://github.com/Yelp/MOE`). The estimated parameter values are saved in "`estimated_parameter.csv`".

2 Model-based Estimates

All the measures below are mean-average of 2,000 simulations with the estimated parameters.

2.1 Trial-by-trial estimate

Trial-by-trial estimates are recorded in "`trial_by_trial_estimate.csv`." Along with the trial-by-trial data from the experiment, a csv file contains model-based measures. Four out of six measures are computed for two phases: a before-feedback phase, where participant saw only pairs of cues; and a after-feedback phase, where participant saw category and exemplar labels but before assigning an object to a cluster. The six measures are described below with column names in the csv file in parentheses.

1. Probability of making a correct response, based on a pair of cues (`p_correct`).
2. Uncertainty in assigning an object to a cluster,
   (a) the entropy of $p(z_i \mid x_i)$ (`representational_uncertainty_before_feedback`); and
   (b) the entropy of $p(z_i \mid x_i,\, y_i,\, y_i')$; and (`representational_uncertainty_after_feedback`).

   This can be thought of as uncertainty about how clearly an object fits in with current knowledge. Here, an estimate close to 0 indicates that an object fits in with current knowledge with certainty.

   In previous work, we [Brad] found anterior hippocampus tracked this measure. This is not the same thing as a (inverse) familiarity/recognition strength.
3. Strength of the best matching cluster.
   (a) the maximum value of $p(z_i \mid x_i)$ (`representational_strength_before_feedback`); and
   (b) the maximum value of $p(z_i \mid x_i,\, y_i,\, y_i')$ (`representational_strength_after_feedback`).

   This captures how close is the best match in memory. Rather than a global memory match signal, this is based on only the best match. Do people retrieve some single best matching representation of the current stimulus?
4. Recognition strength, $p(x_i)$:
   (a) (`recognition_strength_before_feedback`); and
   (b) (`recognition_strength_after_feedback`).

   This measure is akin to global match, familiarity, recognition strength. In the past, we [Brad] have seen posterior hippocampus track this.
5. Probability of new representation,
   (a) (`p_new_representation_before_feedback`); and
   (b) (`p_new_representation_after_feedback`).

This is a probability of forming a new cluster to accommodate an object. This is a new measure that is pretty cool. It's the probability that the current object is stored separately in memory, as opposed to linked/consolidated with an existing representation in memory. The latter promotes learning in the model.

6. Information gained in a trial (`information_gain`): This is KL divergence between $p(z_i \mid x_i, y_i, y_i')$ and $p(z_i \mid x_i)$. This measure gets at surprise from learning and error correction.

2.2 Similarity estimate for RSA

For the RSA measures, there are two sensible ways to compute the matrices, so we computed both. I [Brad] am not sure there is a reason to favor one over the other. Hopefully, they both do the same thing in analyses more or less. This is similarity between pairs of cues (x_i and x_j below), and category and exemplar labels are not considered in this measure. They are stored in "`similarity.csv`."

1. Vector cosine between $p(z_i \mid x_i)$ and $p(z_j \mid x_j)$ (`vector_cosine`); and
2. Probability that x_i and x_j are assigned to the same cluster (`p_same_representation`).