Categorization models

Adaptive network (Rescorla-Wagner)

Restricted to linear predictions

Configural units: interaction terms among cues; enables nonlinearity

Prototype

Based on mean stimulus for each category

$P_c = \frac{1}{n_C}\sum_{\mathbf{x}\in C}\mathbf{x}$

Prediction based on nearest prototype: $\mathrm{argmin}_C \|\mathbf{x} - P_C\|^2$

Also restricted to linear predictions

Exemplar models

Use similarity, not features: $\mathrm{sim}(x, x')$

Similarity often derived from some dimensional representation, e.g. $\mathrm{sim}(x, x') = e^{-\sum_j \alpha_j \left(x_j - x_j'\right)^2}$

Prediction based on summed similarity: $V_C(x) = \sum_{x'\in C} \mathrm{sim}(x, x')$

Can learn any category structure

Nonparametric density estimation

Example: 5/4 structure

Learnable weights: $V_C(x) = \sum_i \beta_{iC}\mathrm{sim}(x, x_i)$

Context model (Medin & Schaffer, 1978; Nosofsky, 1986), static weights: $\beta_{iC} = I_{\{x_i\in C\}}$

ALCOVE (Kruschke, 1992) updates weights for all exemplars: $\Delta\beta_{iC} = \varepsilon\delta_C\mathrm{sim}(x, x_i)$

Pearce (1987, 1994) updates weight only for current stimulus (x_j): $\Delta\beta_{jC} = \varepsilon\delta_C$

Applies in other tasks, e.g. RL:

$Q(a, s) = \sum_{a',s'} \beta_{a',s'}\mathrm{sim}(s, s')\mathrm{sim}(a, a')$

Prototypes, Rescorla-Wagner, and linear regression

RW approximates linear regression

Sequential effects, recency bias

Converges to OLS solution as $\varepsilon \to 0, n \to \infty$

Gradient descent on $\mathrm{E}[(\mathbf{xw} - y)^2]$

Prototype solution

$\|\mathbf{x} - P_A\|^2 < \|\mathbf{x} - P_B\|^2 \Leftrightarrow \langle \mathbf{x}, P_A - P_B\rangle > \langle \frac{P_A + P_B}{2}, P_A - P_B\rangle$ ($\langle\cdot,\cdot\rangle$ denotes inner product)

OLS solution

$\hat{y} = \mathbf{x}(\mathrm{X}^\mathrm{T}\mathrm{X})^{-1}\mathrm{X}^\mathrm{T}Y$

Factorize $n(\mathrm{X}^\mathrm{T}\mathrm{X})^{-1} = \mathrm{UU}^\mathrm{T}$ by positive-definiteness

Yields orthonormal representation $\mathrm{Z} = \mathrm{XU}$:

$\frac{1}{n}\mathrm{Z}^\mathrm{T}\mathrm{Z} = I$

$\hat{y} = \frac{1}{n}\mathbf{z}\mathrm{Z}^\mathrm{T}Y = \langle \mathbf{z}, \frac{n_A}{n}P_A^\mathrm{Z} - \frac{n_B}{n}P_B^\mathrm{Z}\rangle$ (coding Y as ± 1)

→ Prototype model ≃ OLS regression, ignoring cue co/variance and category sizes

Could account for category size by $\langle \mathbf{x}, \frac{n_A}{n}P_A - \frac{n_B}{n}P_B\rangle > \langle \frac{n_A}{n}P_A + \frac{n_B}{n}P_B, \frac{n_A}{n}P_A - \frac{n_B}{n}P_B\rangle$

Feature-similarity equivalence

Feature model predictions depend only on inner products between past and present stimuli

Rescorla-Wagner

After m trials, $\mathbf{w} = \sum_{t=1}^{m} \varepsilon\delta_t\mathbf{x}_t$

Prediction for new stimulus $\mathbf{x}$ is $\langle \mathbf{x}, \mathbf{w}\rangle = \sum_{t=1}^{m} \varepsilon\delta_t\langle \mathbf{x}, \mathbf{x}^t\rangle$

Prototype

Prediction depends on $\langle \mathbf{x}, P_A - P_B \rangle = \sum_{\mathbf{x}_t \in A} \frac{1}{n_A} \langle \mathbf{x}, \mathbf{x}_t \rangle - \sum_{\mathbf{x}_t \in B} \frac{1}{n_B} \langle \mathbf{x}, \mathbf{x}_t \rangle$

or if accounting for category size: $\frac{1}{n} \sum_t y_t \langle \mathbf{x}, \mathbf{x}_t \rangle$

Define $\mathrm{sim}(x, x') = \langle \mathbf{x}, \mathbf{x}' \rangle$

Feature model predictions expressible as $\sum_{t=1}^{m} \beta_t \mathrm{sim}(x, x^t)$

RW corresponds exactly to Pearce update rule: $\Delta\beta_{x_t} = \varepsilon \delta_t$

Prototype model essentially matches context model: $\frac{1}{n}(V_A(x) - V_B(x))$

<u>Similarity-feature equivalence</u>

Any well-behaved similarity function can be written as an inner product

Positive-definite: given $\{x_1, \ldots, x_n\}$, matrix $\left[\mathrm{sim}\left(x_i, x_j\right)\right]$ is positive-semidefinite

Mercer's theorem

Exists (generally infinite) set of features $\mathcal{F} = \{f_i : \mathcal{X} \to \mathbb{R}\}$

Identify x with $\mathbf{x} = \left(f_i(x)\right)_i$

$\mathrm{sim}(x, x') = \sum_i f_i(x) f_i(x') = \langle \mathbf{x}, \mathbf{x}' \rangle$

Important: often sim is defined by some stimulus dimensions; $\mathcal{F}$ will be very different from these

<u>Duality</u>

Feature and similarity models are equivalent at computational level

Given sim model, can get feature model

Given feature model, can get sim model

Different processes on complementarily different representations

Cynical: nonidentifiable

Optimistic: two lenses on same system, with complementary advantages