Hilbert space

Generalizes Euclidean space to possibly infinite dimensions
Useful for studying spaces of functions, sequences, etc.

Real vector space

Addition: \( f + g \in \mathcal{H} \) for all \( f, g \in \mathcal{H} \)
Scalar multiplication: \( rf \in \mathcal{H} \) for all \( f \in \mathcal{H} \) and \( r \in \mathbb{R} \) (or \( r \in \mathbb{C} \) for complex vector space)

Inner product

Generalizes dot product
\( \langle f, g \rangle \in \mathbb{R} \) for all \( f, g \in \mathcal{H} \)
Symmetric: \( \langle f, g \rangle = \langle g, f \rangle \)
Bilinear: \( \langle f_1 + f_2, g \rangle = \langle f_1, g \rangle + \langle f_2, g \rangle \) and \( \langle rf, g \rangle = r \langle f, g \rangle \)
Positive definite: \( \langle f, f \rangle \geq 0 \) and \( \langle f, f \rangle = 0 \rightarrow f = 0 \)

Orthonormal basis
Set of elements \( \{ f_i \} \subset \mathcal{H} \), possibly infinite
Orthonormal: \( \langle f_i, f_j \rangle = 1 \) and \( \langle f_i, f_j \rangle = 0 \) for \( i \neq j \)
Coordinate representation: any \( f \in \mathcal{H} \) can be written (uniquely) as \( f = \sum_i c_i f_i \), with \( c_i = \langle f, f_i \rangle \)
Note that \( \sum_i c_i^2 = \langle f, f \rangle < \infty \)
Explicit characterization of full space: \( \mathcal{H} = \{ \sum_i c_i f_i \mid \sum_i c_i^2 < \infty \} \)
Basis is not unique—can always rotate

Hilbert space of functions learnable on some stimulus set
Expected reward, category membership probability, etc.
Inner product will reflect stimulus representation, enabling link between feature and similarity models

Feature representations

Each stimulus represented as vector of feature values, \( (x_i) \)

Feature set could be finite, \( i \in \{1, ..., m\} \), or infinite, \( i \in \mathbb{N} \)
Assume features are linearly independent (non-redundant): \( \forall x (\sum_i c_i x_i = 0) \leftrightarrow \forall i (c_i = 0) \)

Let \( \mathcal{H} \) be set of all linear functions \( f: \mathcal{X} \rightarrow \mathbb{R} \) based on the features

Linear combinations of feature functions: \( f_i(x) = x_i, \mathcal{H} = \{ \sum_i w_i f_i \mid \sum_i w_i^2 < \infty \} \)
Corresponds to all weight vectors: \( \mathcal{H} = \{ \langle w, \cdot \rangle \mid \sum_i w_i^2 < \infty \} \)
Also expressible as inner products with stimuli (guaranteed by non-redundancy assumption)
\( f \in \mathcal{H} \rightarrow \exists x_1, ..., x_i, c_i, ..., \in \mathbb{R} (f = \sum_i c_i \langle x_i, \cdot \rangle, \sum_i c_i^2 < \infty) \)

Inner product on \( \mathcal{H} \)
Features form orthonormal basis: \( \langle f_i, f_j \rangle = 1 \) and \( \langle f_i, f_j \rangle = 0 \) for \( i \neq j \)
Mirrors inner product on stimuli: let \( f_x = \langle x, \cdot \rangle = \sum_i x_i f_i \), then \( \langle f_x, f_x' \rangle = \langle x, x' \rangle \)

Uniqueness
Features fully determine the Hilbert space
Hilbert space determines the features up to rotation
Rotation (change of coordinate system) has no effect on inner product

Similarity representation

Kernel or similarity function \( k: \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \), with \( k(x, y) = k(y, x) \)
Each stimulus determines an evaluation function \( f_x \), with \( f_x(y) = k(x, y) = f_y(x) \)

Let \( \mathcal{H} \) be set of all functions learnable by summed similarity
\( \mathcal{H} = \{ \sum_i c_i f_{x_i} \mid x_i \in \mathcal{X}, \sum_i c_i^2 < \infty \} \)
Real vector space (addition, scalar multiplication)
Define inner product from kernel
\[ \langle f_x, f_{x'} \rangle = k(x, x') \]
Extend by bilinearity: \[ \langle \sum_i c_i f_{x^i}, \sum_j c'_j f_{y^j} \rangle = \sum_{ij} c_i c'_j k(x^i, y^j) \]
Implies reproducing property: \( f, f_x = f(x) \)
Inner product is positive definite iff \( k \) is

Given stimuli \( x^1, \ldots, x^n \), define kernel matrix \( K \) by \( K_{ij} = k(x^i, x^j) \)
Positive-definite kernel: \( K \) always positive semidefinite, \( \forall b \in \mathbb{R}^n (b^T K b \geq 0) \)

Choose an orthonormal basis \( \{ f_i \} \) for \( \mathcal{H} \)
Guaranteed because \( \mathcal{H} \) is a Hilbert space
Can treat as feature functions
Identify any stimulus \( x \) with vector of feature values, \( x = (f_i(x)) \)

Kernel duality
Dual representations
\[ k(x, x') = \langle x, x' \rangle \]
Same space of learnable functions:
\[ f_x(y) = k(x, y) = \langle x, y \rangle = f_k(y) \text{ so } f_x = f_k \]
\( \mathcal{H} = \{ \sum_i c_i f_{x^i} \mid x^i \in \mathcal{X}, \sum_i c_i^2 < \infty \} = \{ \sum_i w_i f_i \mid \sum_i w_i^2 < \infty \} \)
Equivalence of predictions
Feature model
Knowledge state: feature weights \( w_i \)
Prediction for new stimulus \( x \): \( \langle w, x \rangle \)
Exemplar model
Knowledge state: exemplars \( x^i \) and exemplar weights \( c_i \)
Prediction for new stimulus \( x \): \( \sum_i c_i k(x^i, x) = \langle \sum_i c_i x^i, x \rangle \)
Same predictions iff \( w = \sum_i c_i x^i \)
Equivalence of learning rules
Rescorla-Wagner: \( \Delta w = \varepsilon \delta x_t \)
Equivalent exemplar update rule
Let \( i_t \) be index of exemplar on trial \( t \)
\( \Delta c_{i_t} = \varepsilon \delta \), i.e., update weight only for the presented stimulus
Exactly Pearce’s (1987, 1994) configural learning model

Regularization
Discourage overfitting by modifying objective function with penalty for complexity of solution
Ridge regression
Extend ordinary least squares regression by penalizing sum of squared weights (L2 penalty)
Loss function: \( \mathcal{L}(w) = \sum_i (x^i w - y_i)^2 + \lambda \sum_j w_j^2 = \|Xw - Y\|^2 + \lambda \|w\|^2 \)
Optimal weights: \( \widehat{w} = (X^T X + \lambda I_m)^{-1} X^T Y \)
Bayesian interpretation
Gaussian prior on weight vector: \( w \sim \mathcal{N}(0, \sigma^2 \delta I_m) \)
Data generated as in OLS regression: \( X \sim \mathcal{N}(Xw, \sigma^2 \varepsilon I_n) \)
Posterior: \( p(w \mid X, Y) \propto \exp \left( -\frac{1}{2} \left[ \sigma^2 \varepsilon \|Xw - Y\|^2 + \sigma^2 \delta \|w\|^2 \right] \right) \)
Ridge loss function is (linear transform of) log-posterior with \( \lambda = \sigma^2 \varepsilon / \sigma^2 \delta \)
Other penalties, e.g. Lasso (L1 regularization): \( \lambda = \sum_j |w_j| \)
Gaussian process regression

Rewrite ridge estimate as \( \hat{\omega} = X^T (XX^T + \lambda I_n)^{-1} Y \)

\( n \times n \) matrix instead of \( m \times m \); more efficient if fewer cases than cues

Prediction for new stimulus: \( \hat{y} = x X^T (XX^T + \lambda I_n)^{-1} Y \)

Depends only on inner products between stimuli

\[ \hat{y} = K(x_{\text{test}}, x_{\text{train}})(K(x_{\text{train}}, x_{\text{train}}) + \lambda I_n)^{-1} y_{\text{train}} \]

Bayesian interpretation

Random function \( f: \mathcal{X} \rightarrow \mathbb{R} \)

Jointly Gaussian for any finite set of stimuli: \( (f(x^1), ..., f(x^n))^T \sim \mathcal{N}(0, K) \) where \( K_{ij} = k(x^i, x^j) \)

Noisy observations: \( y \sim \mathcal{N}(f(x), \lambda) \)

Same generative model as ridge regression with \( \sigma_n^2 = 1 \): \((x^1 w, ..., x^n w)^T = X w \sim \mathcal{N}(0, XX^T)\)

Posterior prediction

\[
p(f_{\text{all}} | X_{\text{train}}, Y_{\text{train}}) \propto \exp \left( -\frac{1}{2} (f_{\text{all}}^T K_{\text{all}}^{-1} f_{\text{all}} + \lambda^{-1} (f_{\text{train}} - y_{\text{train}})^T (f_{\text{train}} - y_{\text{train}})) \right)
\]

\[
\hat{f}_{\text{all}} = \left( K_{\text{all}}^{-1} + \lambda^{-1} \left[ I_{n_{\text{train}}} \ 0 \right] \right)^{-1} \left[ I_{n_{\text{train}}} \ 0 \right] y_{\text{all}}
\]

\[
= K_{\text{all}, \text{train}} (K_{\text{train}, \text{train}} + \lambda I_{n_{\text{train}}})^{-1} y_{\text{train}}
\]