A function MRI mind-reading game

Charles Zheng and Yuval Benjamini

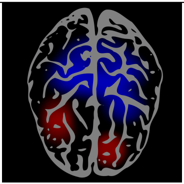
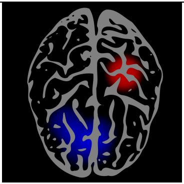
Stanford University

March 28, 2015

Section 1

Introduction

Functional MRI

Stimuli	Response
	
	

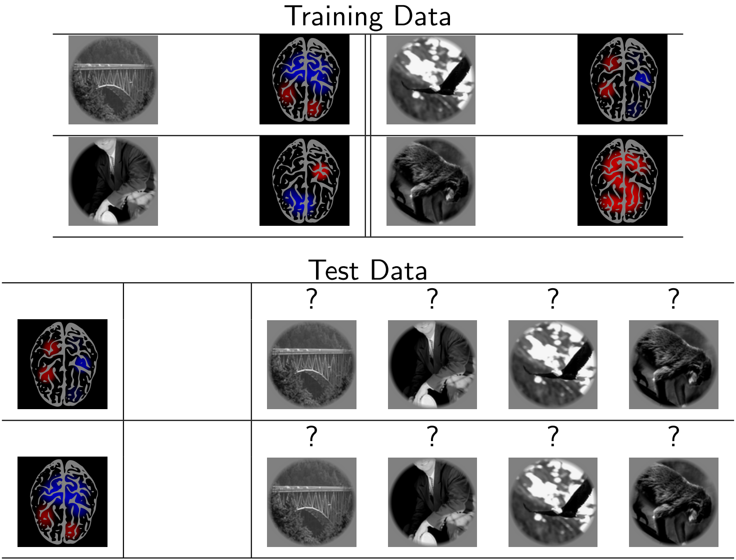
Functional MRI

Stimuli x	Response y
$\begin{pmatrix} 1.0 \\ 0 \\ 3.0 \\ 0 \\ -1.2 \end{pmatrix}$	$\begin{pmatrix} 1.2 \\ 0 \\ -1.8 \\ -1.2 \end{pmatrix}$
$\begin{pmatrix} 0 \\ -2.2 \\ -3.1 \\ 4.5 \\ 0 \end{pmatrix}$	$\begin{pmatrix} -1.2 \\ -1.9 \\ 0.5 \\ 0.6 \end{pmatrix}$

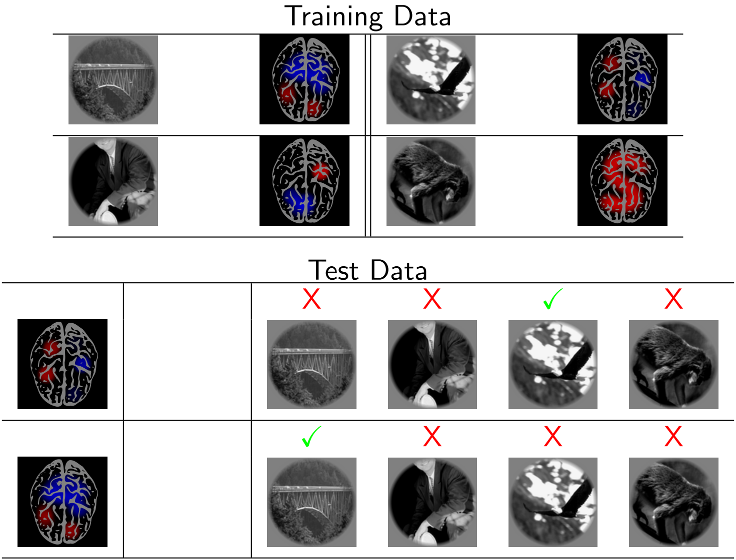
Encoding vs Decoding

- Encoding: predict y from x.
- Decoding: reconstruct x from y (mind-reading).

A mind-reading game: Classification

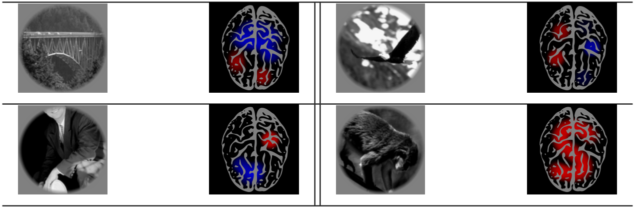


A mind-reading game: Classification

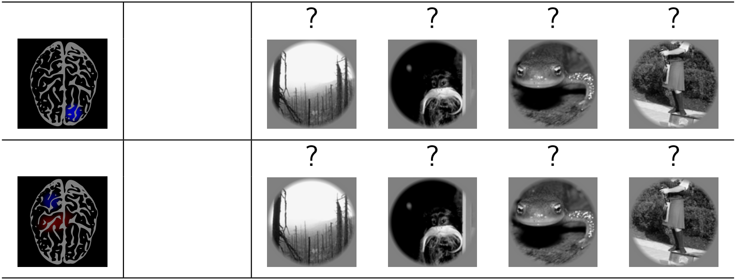


A mind-reading game: Identification

Training Data



Test Data *(new images!)*



Statistical formulation I

Training data.

- Given training classes $S_{\text{train}} = \{\text{train:}1, \ldots, \text{train:}k\}$ where each class train:i has features $x_{\text{train:}i}$.
- For $t = 1, \ldots, T_{\text{train}}$, choose class label $z_{\text{train:}t} \in S_{\text{train}}$; generate

$$y_{\text{train:}t} = f(x_{z_{\text{train:}t}}) + \epsilon_t$$

  where f is an unknown function, and ϵ_t is i.i.d. from a known or unknown distribution.

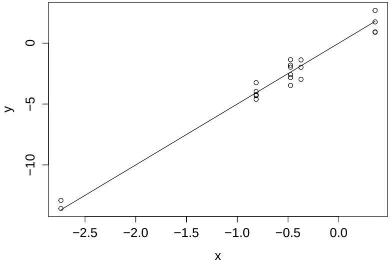
Test data.

- Given test stimuli $S_{\text{test}} = \{\text{test:}1, \ldots, \text{test:}\ell\}$ with features $\{x_{\text{test:}1}, \ldots, x_{\text{test:}\ell}\}$
- Task: for $t = 1, \ldots, T_{\text{test}}$, predict $\hat{z}_{\text{test:}t} \in S_{\text{train}}$ from $y_{\text{test:}t}$; try to minimize misclassification rate

Statistical formulation II

- f is an unknown function
- P is a known or unknown distribution over image features
- *Training data.* Draw $x_{\text{train}:i} \sim P$ for $i = 1\, hdots, k$.
- *Test data.* Draw $x_{\text{train}:i} \sim P$ for $i = 1\, hdots, \ell$.
- Theoretical question: Analyze average misclassification rate when classes are generated this way

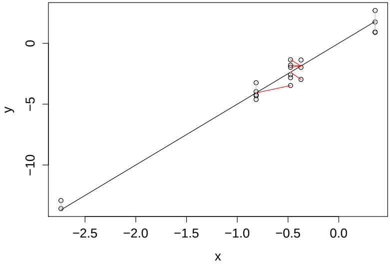
Toy example I



Model.

- Features x are one-dimensional real numbers, as are responses y. Parameter β is also a real number.
- Model is linear: $y \sim N(x\beta, \sigma_\epsilon^2)$
- Suppose we estimated $\hat{\beta}$ from training data.

Toy example I: Test data



- Generate features $x_{\text{test}:1}, \ldots, x_{\text{test}:\ell}$ iid $N(0, \sigma_x^2)$.
- Hidden labels $z_{\text{test}:t}$ are iid uniform from S_{train}. Generate $y_{\text{test}:t} \sim N(\beta x_{z_{\text{test}:t}}, \sigma_\epsilon^2)$
- Estimate $\hat{z}_{\text{test}:t}$ by maximum likelihood assuming $\hat{\beta}$ is correct. Thus:

$$\hat{z}_{\text{test}:t} = \text{argmin}_z (\hat{\beta} x_z - y_{\text{test}:t})^2$$