


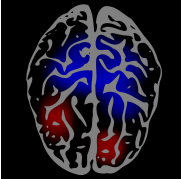

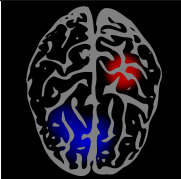
# A functional MRI mind-reading game

Charles Zheng and Yuval Benjamini

Stanford University

April 2, 2015

# Functional MRI

Stimuli	Response
	
	

# Functional MRI

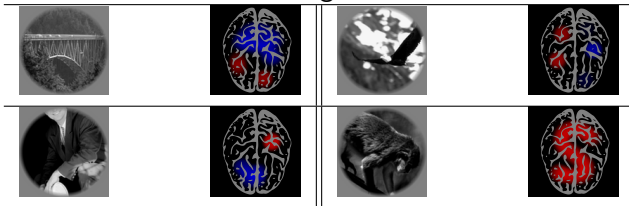
Stimuli $x$	Response $y$
$\begin{pmatrix} 1.0 \\ 0 \\ 3.0 \\ 0 \\ -1.2 \end{pmatrix}$	$\begin{pmatrix} 1.2 \\ 0 \\ -1.8 \\ -1.2 \end{pmatrix}$
$\begin{pmatrix} 0 \\ -2.2 \\ -3.1 \\ 4.5 \\ 0 \end{pmatrix}$	$\begin{pmatrix} -1.2 \\ -1.9 \\ 0.5 \\ 0.6 \end{pmatrix}$

# Encoding vs Decoding

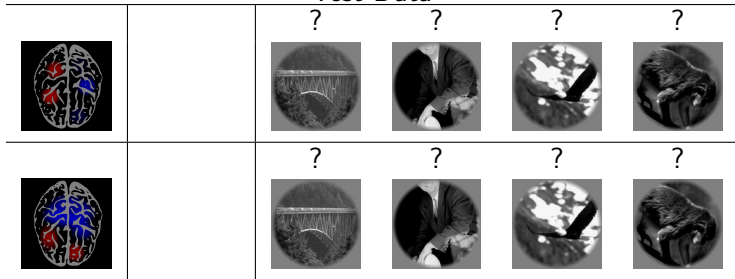
- Encoding: predict  $y$  from  $x$ .
- Decoding: reconstruct  $x$  from  $y$  (mind-reading).
  - Classification: label response  $y$  by a class from the training data
  - Identification: label response  $y$  by a class *outside* of the training data
  - Reconstruction: infer  $x$  from  $y$

# Classification

Training Data

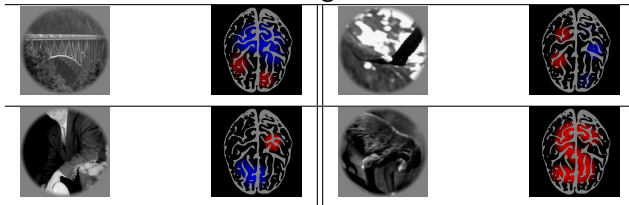


Test Data

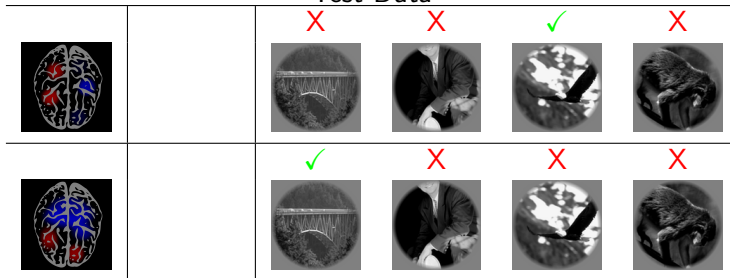


# Classification

Training Data

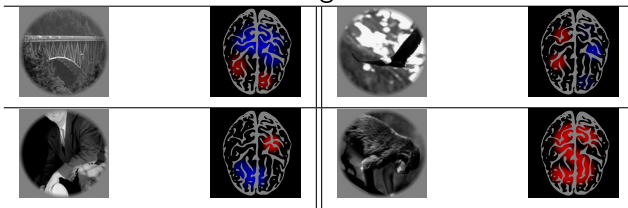


Test Data

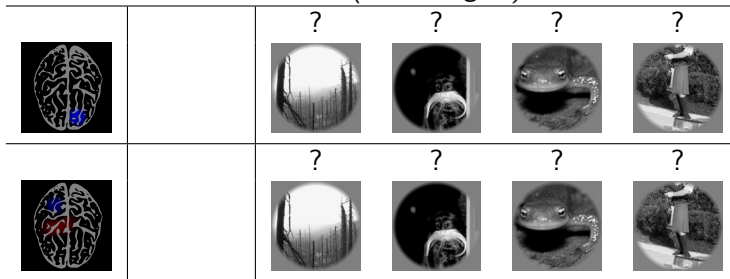


# Identification

Training Data

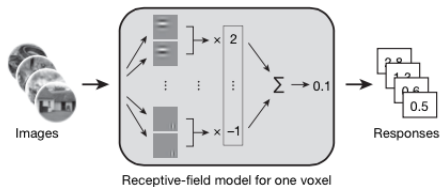


Test Data (*new images!*)



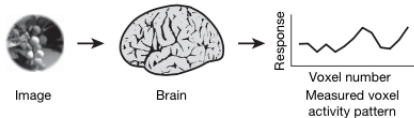
### Stage 1: model estimation

Estimate a receptive-field model for each voxel

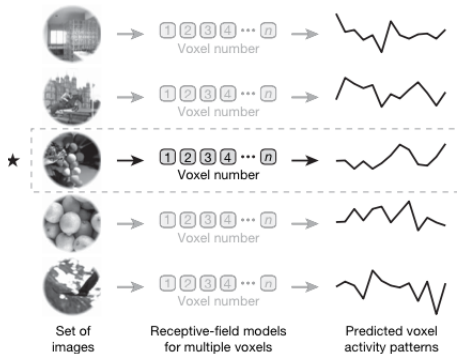


### Stage 2: image identification

(1) Measure brain activity for an image



(2) Predict brain activity for a set of images using receptive-field models

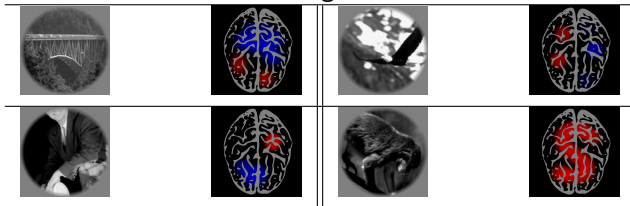


(3) Select the image (★) whose predicted brain activity is most similar to the measured brain activity

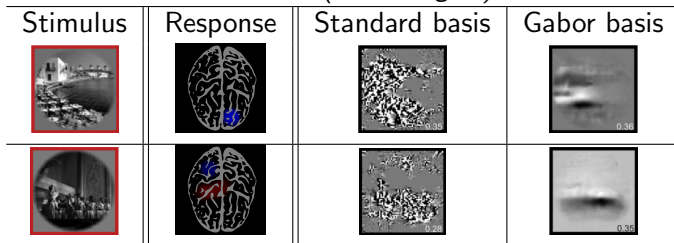


# Reconstruction

Training Data



Test Data (*new images!*)



# Classification vs Identification vs Reconstruction

- Classification is easy: doesn't require domain-specific model
- Identification and reconstruction both require a model relating image features to responses

## *Difficulty of Identification vs Reconstruction*

	High dimensions	Number of candidate stimuli
Identification	Neutral	Hard
Reconstruction	Hard	Easy

# Motivating questions

- Under what conditions would it be possible to get performance on reconstruction or identification?
- How can we develop methods which achieve better performance on these tasks?
- Can we interpret the performance metric (prediction error, misclassification error) of a model to draw scientific conclusions? (E.g. which features are important, information content of fMRI scan.)

# Classification vs Identification vs Reconstruction

## *Supervised learning problems*

	Classification	Identification	Regression
<b>Class label</b>	$z$	$z$	$i = 1, \dots, n$
Observed in training	Yes	Yes	Yes
Observed in test	No	No	Yes
<b>Class feature</b>	$x_z$	$x_z$	$x_i$
Observed in training	No	Yes	Yes
Observed in test	No	No	Yes
<b>Response</b>	$y$	$y$	$y$
Observed in training	Yes	Yes	Yes
Observed in test	Yes	Yes	No

- Unified interpretation of supervised learning problems
- Class  $z$  has feature  $x_z$ ,  $y \sim f_{x_z}$
- Problems differ in terms of observed and unobserved information

# Classification vs Identification vs Reconstruction

## *Supervised learning problems*

	Misclassification Rate	Prediction error
No covariates	Classification	(nothing to predict)
Covariates ( $x$ )	Identification	Regression

- Reconstruction is regression  $x \sim y$
- Does there already exist statistical theory for identification?
- Next: a toy model for identification

## Section 2

# Theory

# The problem of identification

## *Training data.*

- Given training classes  $S_{\text{train}} = \{\text{train}:1, \dots, \text{train}:k\}$  where each class  $\text{train}:i$  has features  $x_{\text{train}:i}$ .
- For  $t = 1, \dots, T_{\text{train}}$ , choose class label  $z_{\text{train}:t} \in S_{\text{train}}$ ; sample a response  $y_{\text{train}:t}$  from that class.

## *Test data.*

- Given test classes  $S_{\text{test}} = \{\text{test}:1, \dots, \text{test}:\ell\}$  with features  $\{x_{\text{test}:1}, \dots, x_{\text{test}:\ell}\}$
- Task: for  $t = 1, \dots, T_{\text{test}}$ , label  $y_{\text{test}:t}$  by class  $\hat{z}_{\text{test}:t} \in S_{\text{train}}$ ; try to minimize misclassification rate

# Additional assumptions

- For a point  $y$  from class with features  $x$ ,

$$y = f(x) + \epsilon$$

where the noise  $\epsilon$  is drawn from some distribution and  $f$  is an unknown function

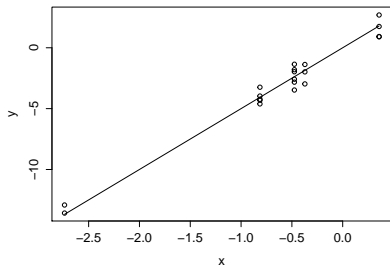
- The features for the training and test classes are sampled iid from the same distribution  $P$

$$x_{\text{train}:i} \sim P$$

$$x_{\text{train}:i} \sim P$$



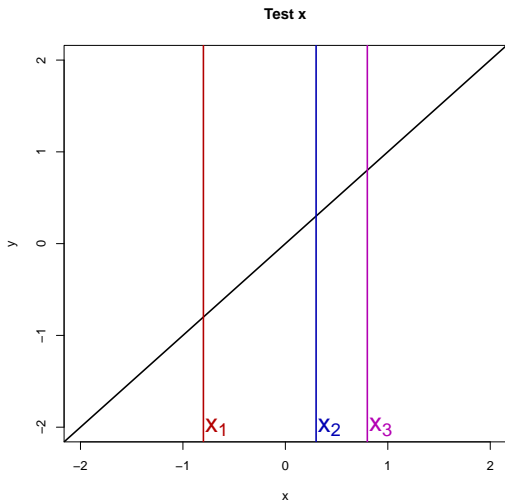
# Toy example I



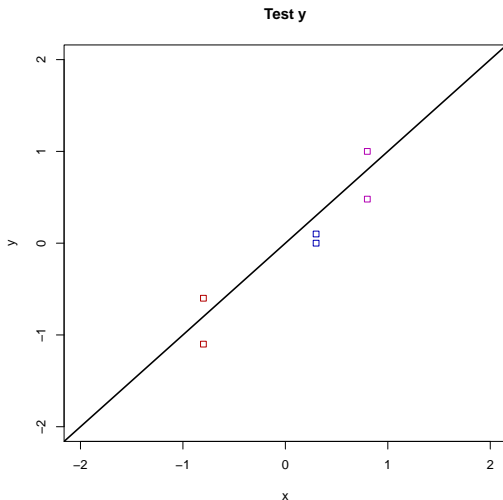
- Features  $x$  are one-dimensional real numbers, as are responses  $y$ . Parameter  $\beta$  is also a real number.
- Model is linear:  $y \sim N(x\beta, \sigma_\epsilon^2)$

The plot shows two linear functions,  $f$  (black line) and  $\hat{f}$  (red line), plotted against  $x$  and  $y$ . Both lines pass through the origin  $(0,0)$ . The black line  $f$  has a steeper slope than the red line  $\hat{f}$ , indicating that the fit  $\hat{f}$  is biased downwards (underestimates  $y$  for a given  $x$ ).

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡ 🔍 ↺



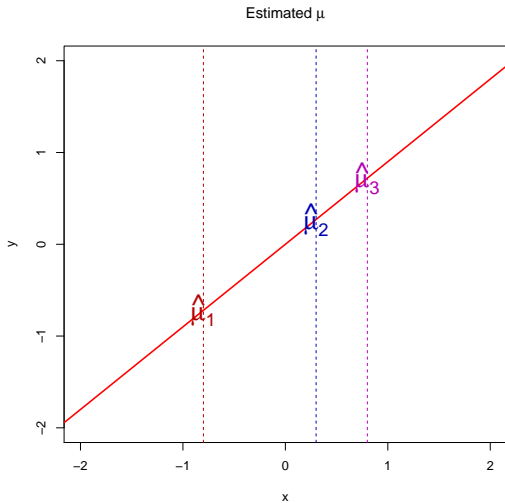
Generate features  $x_{\text{test}:1}, \dots, x_{\text{test}:\ell}$  iid  $N(0, \sigma_x^2)$ .



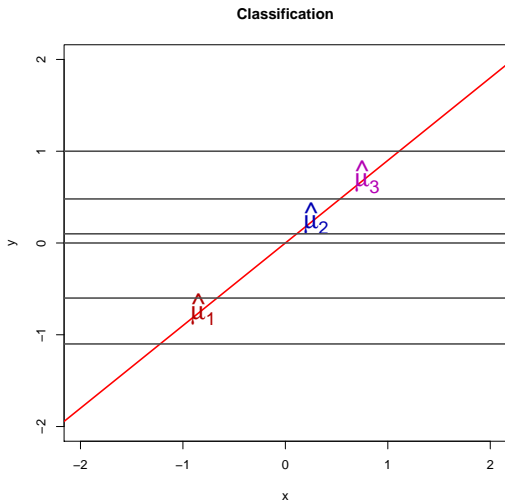
Hidden labels  $z_{\text{test}:t}$  are iid uniform from  $S_{\text{train}}$ .

Generate  $y_{\text{test}:t} \sim N(\beta x_{z_{\text{test}:t}}, \sigma_{\epsilon}^2)$



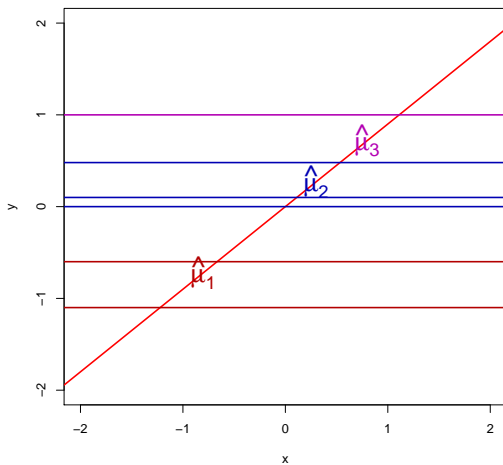


$$\hat{\mu}_{\text{test}:i} = \hat{\beta} x_{\text{test}:i}$$



$$\hat{z}_{\text{test}:t} = \operatorname{argmin}_z \ell_{\hat{\mu}_z}(y_{\text{test}:t})$$

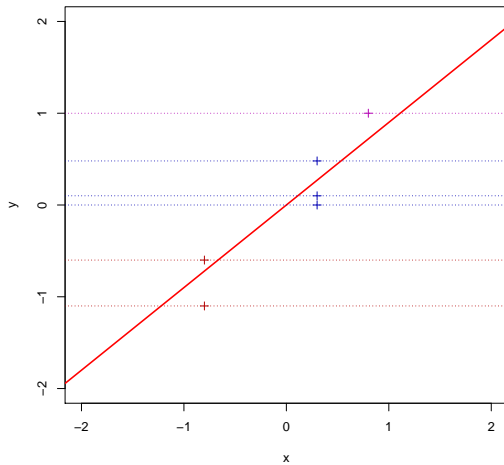
# Classification



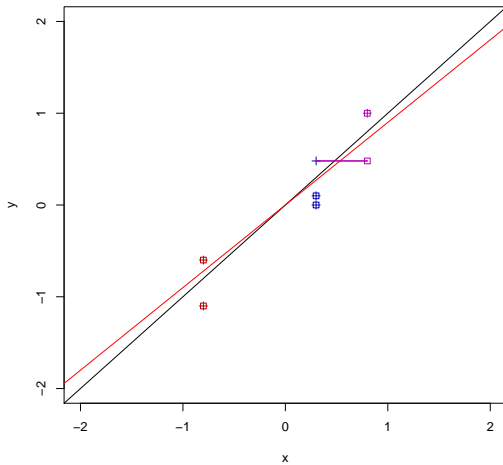
$$\hat{z}_{\text{test}:t} = \operatorname{argmin}_z (\hat{\mu}_z - y_{\text{test}:t})^2$$



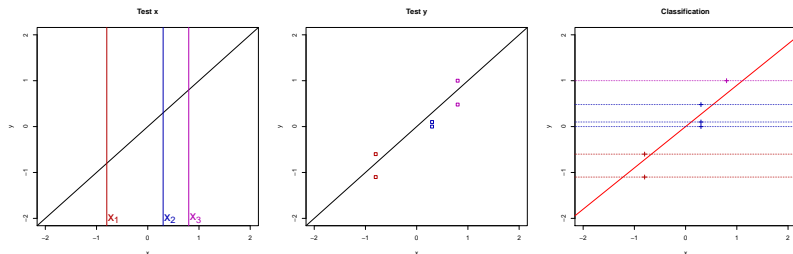
## Classification



### Misclassification



# Toy example I



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

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

# Toy example I: Questions

- 1 We know the prediction error is minimized when  $\hat{\beta} = \beta$ . Is it also true that misclassification error in the mind-reading game is minimized when  $\hat{\beta} = \beta$ ?
- 2 Even if the answer to 1. is yes, should we estimate  $\hat{\beta}$  using the same methods as in least-squares regression?

# Question 1: Outline

We will find an answer to question 1 as follows

- Write an explicit expression for the misclassification rate as a function of  $\hat{\beta}$
- Take the derivative of that expression with respect to  $\hat{\beta}$  at the true  $\beta$
- Does that derivative equal zero?
- If so, look at second derivatives, lower bounds, etc.

*Write an explicit expression for the misclassification rate*

- The expected misclassification error is the same if we take  $T_{\text{test}} = 1$ . Then let  $(x_*, y_*)$  be the feature-response pair in the test set, where

$$y_* = x_*\beta + \epsilon_*$$

- Denote the features for the incorrect classes as  $x_1, \dots, x_{\ell-1}$ .
- Let  $\delta = \hat{\beta} - \beta$ .

*Write an explicit expression for the misclassification rate (cont.)*

- Ignore the possibility of ties. The response  $y_*$  is misclassified if and only if

$$\min_{i=1,\dots,\ell-1} |y_* - x_i \hat{\beta}| < |y_* - x_* \hat{\beta}|$$

equivalently

$$\cup_{i=1,\dots,\ell-1} E_i$$

where  $E_i$  is the event that

$$|y_* - x_i \hat{\beta}| < |y_* - x_* \hat{\beta}|$$

*Write an explicit expression for the misclassification rate (cont.)*

- Use the following conditioning

$$\mathbf{E}[\text{misclassification}] = \mathbf{E}[\mathbf{E}[\Pr_{x_1, \dots, x_\ell}[\cup_i E_i] | x_* = x, \epsilon_* = \epsilon]]$$

- Use the fact that events  $E_i$  are independent and have the same probability, thus:

$$\mathbf{E}[\text{misclassification}] = 1 - \mathbf{E}[\mathbf{E}[(1 - \Pr[E_1])^{\ell-1} | x_* = x, \epsilon_* = \epsilon]]$$

- Next: write an expression for  $\Pr[E_1]$



Write an expression for  $\Pr[E_1]$ .

- $E_1$  can also be written as the event

$$|x_*\beta + \epsilon_* - x_1(\beta + \delta)| < |-\delta x_* + \epsilon_*|$$

- Conditioning on  $\epsilon_*$  and  $x_*$ , we have

$$\Pr[E_1] = \left| \Phi\left(\frac{x_*}{\sigma_x}\right) - \Phi\left(\frac{x_*(\beta - \delta) + 2\epsilon_*}{\sigma_x(\beta + \delta)}\right) \right|$$

An exact expression for expected misclassification is therefore

$$1 - \int_{\epsilon} \left[ \int_x \left( 1 - \left| \Phi\left(\frac{x}{\sigma_x}\right) - \Phi\left(\frac{x(\beta-\delta)+2\epsilon}{\sigma_x(\beta+\delta)}\right) \right| \right)^{\ell-1} d\Phi\left(\frac{x}{\sigma_x}\right) \right] d\Phi\left(\frac{\epsilon}{\sigma_{\epsilon}}\right)$$

*Take the derivative of the expression with respect to  $\delta$*

Fix  $\epsilon > 0$ . The derivative of the inner integral wrt  $\delta = 0$  is proportional to

$$\int_x (1 - \Phi(\frac{x\beta + 2\epsilon}{\sigma_x\beta}) + \Phi(\frac{x}{\sigma_x})) \phi(\frac{x\beta + 2\epsilon}{\sigma_x\beta}) (x + \frac{\epsilon}{\beta}) \phi(\frac{x}{\sigma_x}) dx$$

*Is the derivative zero?*

*Is the derivative zero?*

Note that

$$\phi\left(\frac{x\beta + 2\epsilon}{\sigma_x\beta}\right) \phi\left(\frac{x}{\sigma_x}\right) \propto \phi\left(\frac{\sqrt{2}(x + \frac{\epsilon}{\beta})}{\sigma_x}\right)$$

which is the density of a normal variate with mean  $-\epsilon/\beta$

But now note that the other terms

$$\left(1 - \Phi\left(\frac{x\beta + 2\epsilon}{\sigma_x\beta}\right) + \Phi\left(\frac{x}{\sigma_x}\right)\right) \left(x - \frac{\epsilon}{\beta}\right)$$

are antisymmetric about  $x = -\frac{\epsilon}{\beta}$ .

Thus by symmetry, the derivative of the inner integral  $\delta = 0$  vanishes. The same argument works for  $\epsilon < 0$ , hence the misclassification rate is stationary at  $\hat{\beta} = \beta$ .

*(We'll skip the second derivative checking, etc.)*

# Question 1: Remarks

Optimal  $\hat{\beta} = \beta$

- Obvious for  $\epsilon \sim N(0, \sigma_\epsilon^2)$  no matter the distribution of  $x$ ... but
- $\epsilon$  can have any distribution... if  $x$  is normally distributed
- Don't know for  $\epsilon$  arbitrary and  $x$  arbitrary...

# Toy example I: Estimation

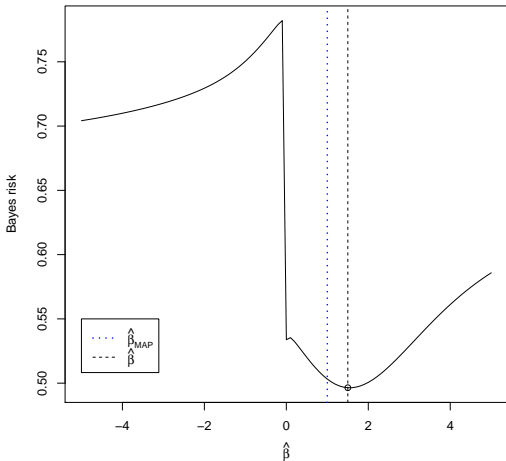
- Second question: what about estimation?
- Take a Bayesian viewpoint: suppose we have a prior distribution for  $\beta$
- For *least-squares regression*, we would use  $\hat{\beta} = \int \beta p_{\text{posterior}}(\beta) d\beta$ , the posterior mean.
- For *identification*, we would choose

$$\hat{\beta} = \operatorname{argmin}_{\hat{\beta}} \int R(\beta; \hat{\beta}) p_{\text{posterior}}(\beta) d\beta$$

where  $R$  is the expected misclassification rate.

- How will these differ?

### Bayes estimation

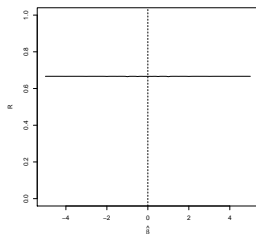


Point estimate for identification (black dashed) is larger than posterior mean (blue dotted)

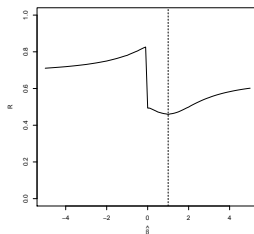
# Toy example I: Estimation

*Why the upward bias?*

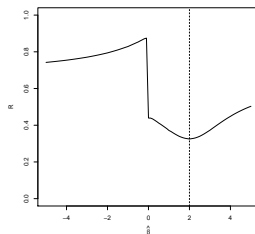
$$\beta = 0$$



$$\beta = 1$$



$$\beta = 2$$



Risk function is more sensitive for large  $\beta$ .



# Estimation: questions

- Is the optimal  $\hat{\beta}$  for identification is in general “larger” than the optimal  $\hat{\beta}$  for regression, in a frequentist (e.g. minimax) sense?
- Lasso/Ridge penalized regression models are commonly used for identification
- Hypothesis: the optimal  $\lambda$  for identifying  $x$  from  $y$  will be smaller (hence produce less sparse  $\hat{\beta}$ ) than the optimal  $\lambda$  for regression  $y \sim x$ .

# Generalizing to higher dimensions

## Model fitting

- $x$  is  $p$ -dimensional column vector,  $y$  is  $q$ -column vector
- Using training data, learn a model

$$y = B^T x + b^T + \epsilon$$

where  $B$  is a  $p \times q$  matrix and  $b$  is a  $q$ -row vector.

- Using residuals from training data, estimate  $\hat{\Sigma}_\epsilon$

## Identification

- For each test class feature  $x_{\text{test}:i}$ , compute the predicted mean response

$$\mu_{\text{test}:i} = B^T x_{\text{test}:i} + b^T$$

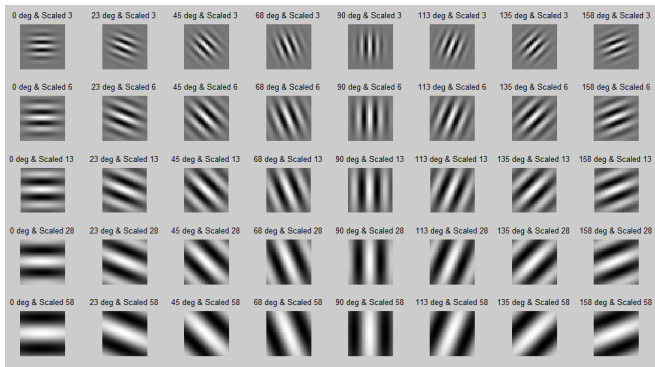
- (MLE) Label a new response  $y_*$  with test class  $z$  that minimizes

$$(\mu_z - y_*)^T \hat{\Sigma}_\epsilon^{-1} (\mu_z - y_*)$$

## Section 3

# Experiments

- From Kay *et al.* paper
- 1750 images with averaged responses from 2 repeats
- Responses  $y$ : 100 selected voxels from the most basic visual subsystem, V1
- Features  $x$ : 10921 image features based on Gabor filters



# Regression vs Identification

## *Partition*

- Randomly partition into training set (1725) and test set (25)

## *Model fitting via lasso*

- Notation:  $Y = (y_{\text{train}:1}, \dots, y_{\text{train}:1725})^T$ ,  $X = (x_{\text{ztrain}:1}, \dots, x_{\text{ztrain}:1725})^T$
- Fix  $\lambda$ . Fit a separate Lasso regression for each voxel:

$$\text{minimize } \frac{1}{2} \|Y_i - \hat{\beta}^{(i)}X + \hat{\beta}_0^{(i)}\|^2 + \lambda \|\hat{\beta}\|_1$$

- Let  $B = (\hat{\beta}^{(1)}, \dots, \hat{\beta}^{(100)})$ ,  $b = (\hat{\beta}_0^{(1)}, \dots, \hat{\beta}_0^{(100)})$

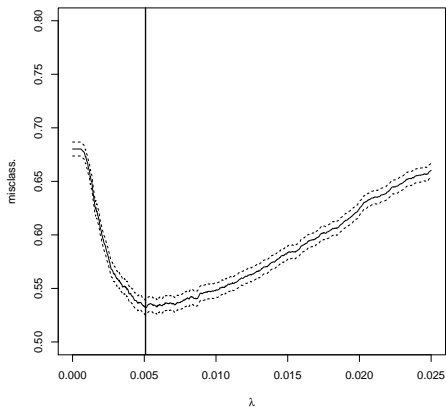
## *Performance on test set*

- Regression: use test labels to predict  $\hat{y}$
- Identification: for test responses  $y$ , estimate label using MLE

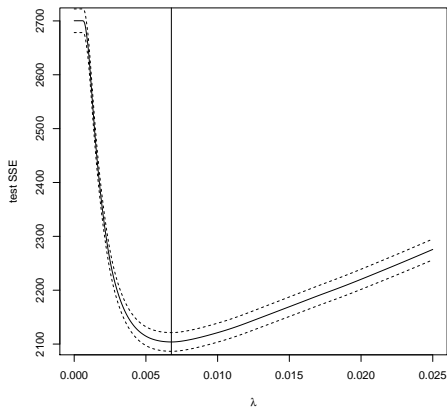
Perform this experiment for  $\lambda \in [0, 0.025]$  for 200 random partitions

# Results

Identification



Regression



Optimal  $\lambda$  for identification is smaller... but difference not significant

## Section 4

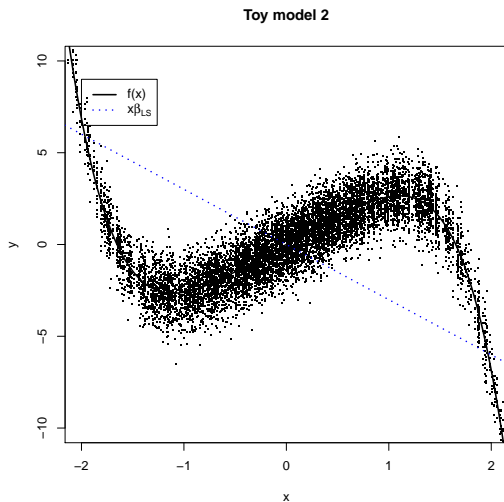
### Nonlinear toy example

# More questions

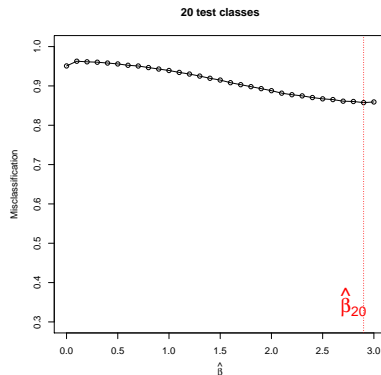
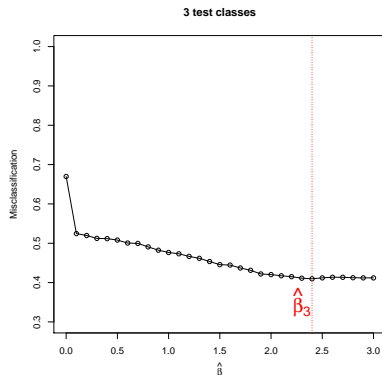
- ③ What happens if the true regression function  $f$  is nonlinear, but we restrict  $\hat{f}$  to be linear?
- ④ What happens when the number of classes  $\ell$  increases? What if  $\ell$  increases while  $\sigma_\epsilon^2$  decreases?



# Toy example IIa

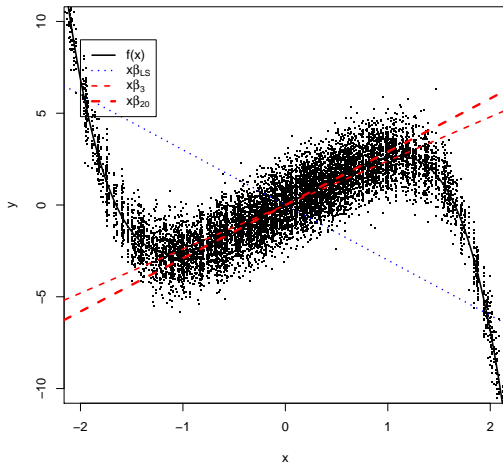


# Toy example IIa



Effect of increasing  $\ell$ .

Toy model 2



# Why is this?

- We can relate identification to regression with a different loss function
- Least squares loss

$$\mathbf{E}[(y - \hat{y})^2]$$

- Identification loss

$$\mathbf{E}[1 - \Pr[|y - \hat{y}'| < |y - \hat{y}|]^{\ell-1}]$$

where  $\hat{y}'$  is the predicted value for a randomly drawn  $x$ .

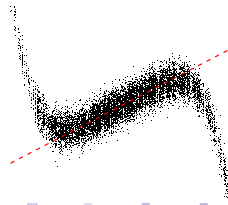
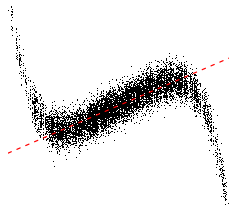
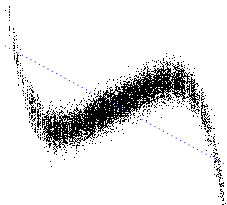
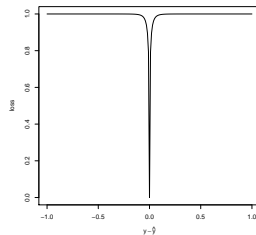
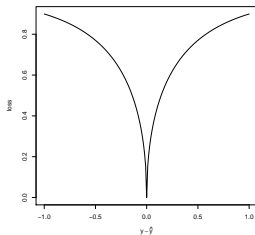
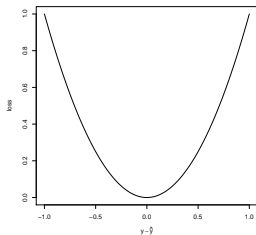
# Why is this?

Identification loss more closely resembles 0-1 loss as  $\ell$  increases.

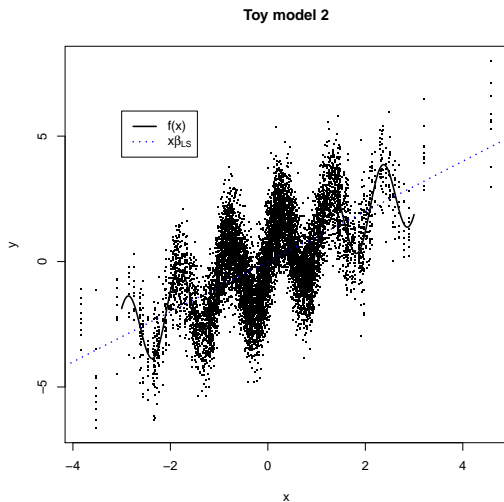
Squared error

$\ell = 3$

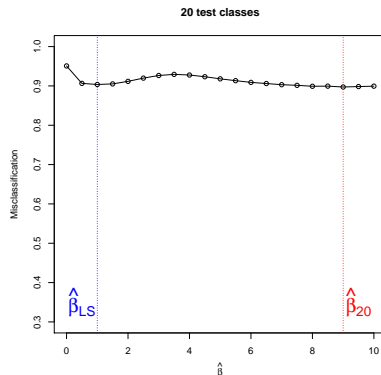
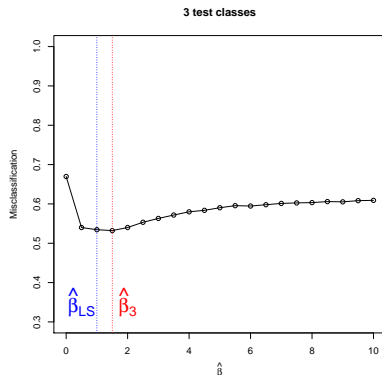
$\ell = 20$



# Toy example IIb

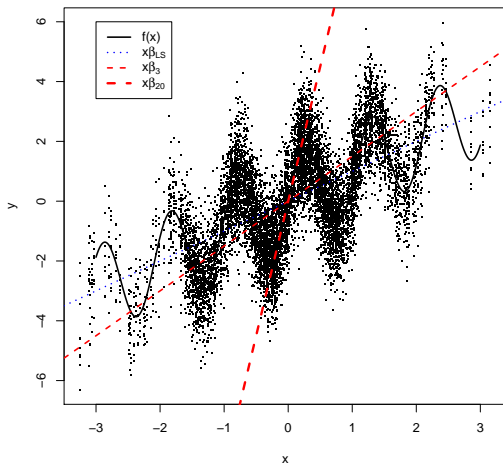


# Toy example IIb



Effect of increasing  $\ell$ .

### Toy model 2



Effect of increasing  $\ell$ : global trends will become ignored in favor of locally linear trends!



# Implications

- “The model is always wrong”
- Statistical methods should be robust to small deviations from the model
- Even when minor nonlinearities exist in the model, identification performance fails to reflect global fit

# Conclusions

- The problem of *decoding*, predicting  $x$  from  $y$ , is of interest to many neuroscientists
- Different formulations of the decoding problem: classification, identification, and reconstruction (regression) have different properties and advantages
- Statistical theory can help with training the models *and* with interpreting the results

*In particular...*

- Identification is similar to regression  $y \sim x$  in a special case, but can benefit from less sparse estimates.
- Identification can lead to counterintuitive results when there are nonlinearities and  $\ell$  is large

- Kay, KN., Naselaris, T., Prenger, R. J., and Gallant, J. L. “Identifying natural images from human brain activity”. *Nature* (2008)
- Naselaris, et al. “Bayesian reconstruction of natural images from human brain activity”. *Neuron* (2009)
- Vu, V. Q., Ravikumar, P., Naselaris, T., Kay, K. N., and Yu, B. “Encoding and decoding V1 fMRI responses to natural images with sparse nonparametric models”, *The Annals of Applied Statistics*. (2011)
- Chen, M., Han, J., Hu, X., Jiang, Xi., Guo, L. and Liu, T. “Survey of encoding and decoding of visual stimulus via fMRI: an image analysis perspective.” *Brain Imaging and Behavior*. (2014)