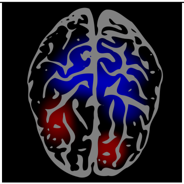
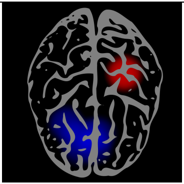
A functional MRI mind-reading game

Charles Zheng and Yuval Benjamini

Stanford University

March 30, 2015

Functional MRI

Stimuli	Response
	
	

Functional MRI

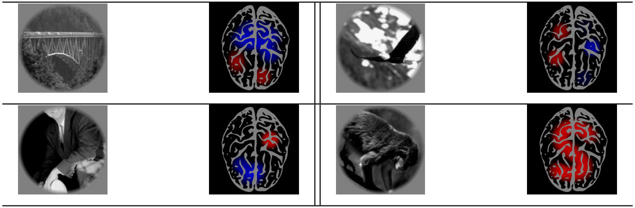
Stimuli x	Response y
$\begin{pmatrix} 1.0 \\ 0 \\ 3.0 \\ 0 \\ -1.2 \end{pmatrix}$	$\begin{pmatrix} 1.2 \\ 0 \\ -1.8 \\ -1.2 \end{pmatrix}$
$\begin{pmatrix} 0 \\ -2.2 \\ -3.1 \\ 4.5 \\ 0 \end{pmatrix}$	$\begin{pmatrix} -1.2 \\ -1.9 \\ 0.5 \\ 0.6 \end{pmatrix}$

Encoding vs Decoding

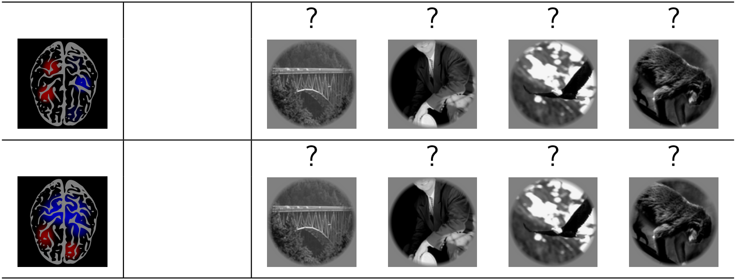
- Encoding: predict y from x.
- Decoding: reconstruct x from y (mind-reading).

A mind-reading game: Classification

Training Data

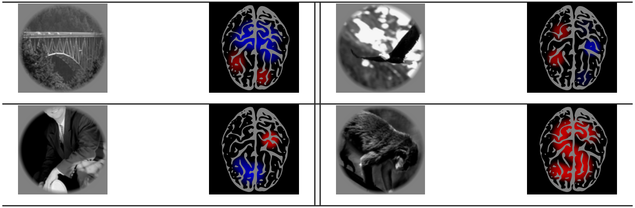


Test Data

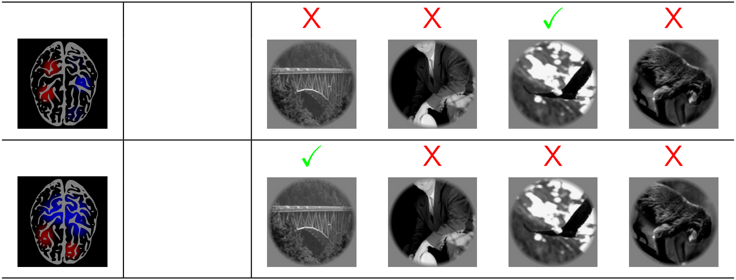


A mind-reading game: Classification

Training Data

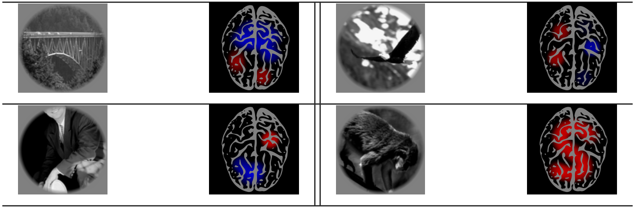


Test Data

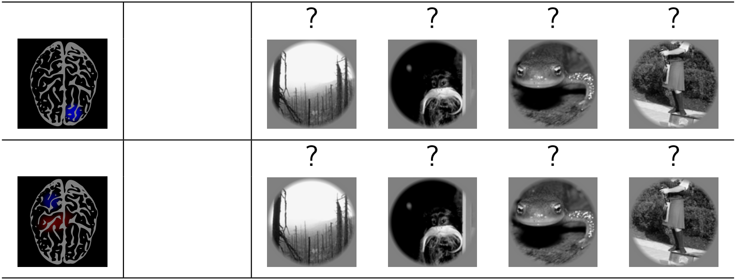


A mind-reading game: Identification

Training Data



Test Data *(new images!)*



Section 2

Theory

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.

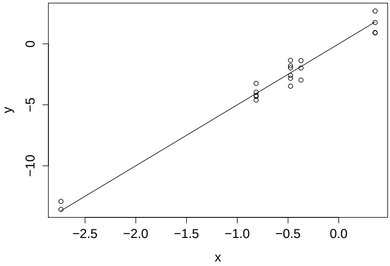
Test data.

- 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 $t = 1, \ldots, T_{\text{test}}$, label $y_{\text{test:}t}$ by stimulus $\hat{z}_{\text{test:}t} \in S_{\text{train}}$; try to minimize misclassification rate

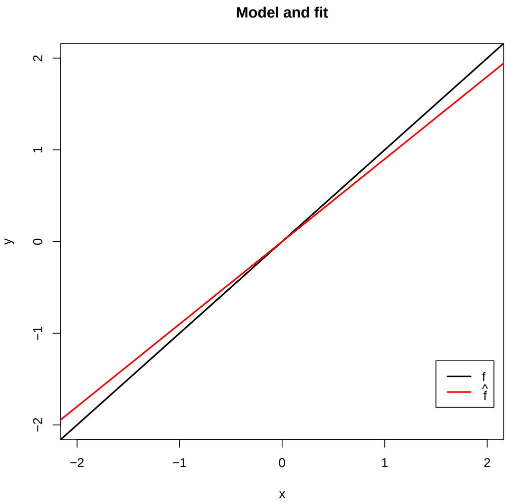
Statistical formulation II

- 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

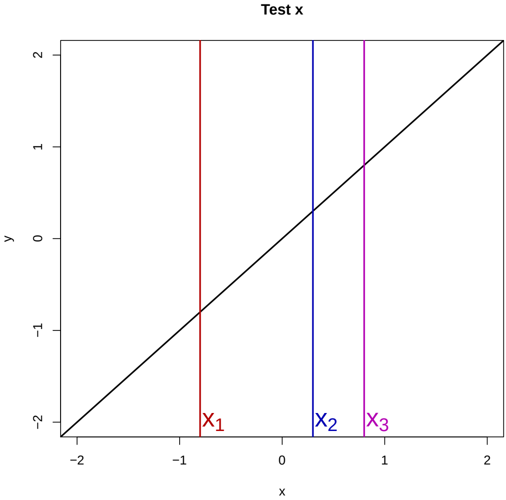
Toy example I



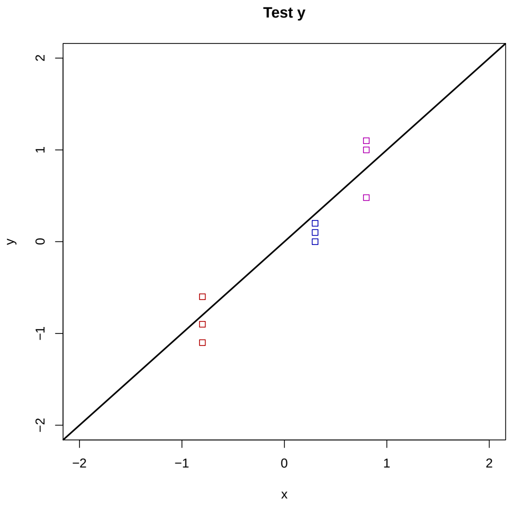
- Features x are one-dimensional real numbers, as are responses y. Parameter β is also a real number.
- Model is linear: $y \sim N(x\beta, \sigma_\epsilon^2)$



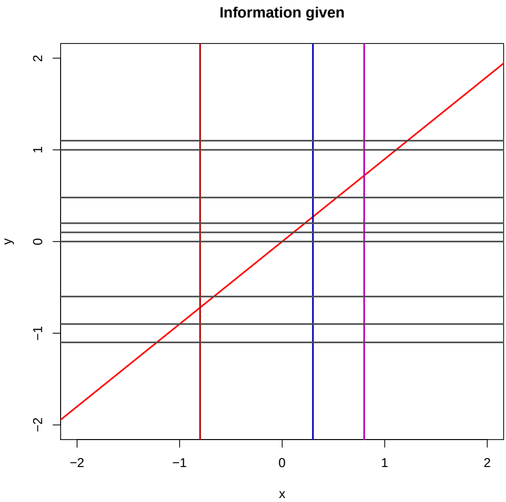
Suppose we estimated $\hat{\beta}$ from training data.



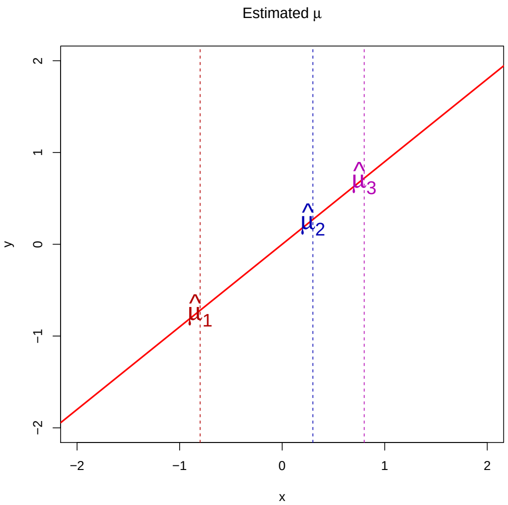
Generate features $x_{\text{test}:1}, \ldots, x_{\text{test}:\ell}$ iid $N(0, \sigma_x^2)$.



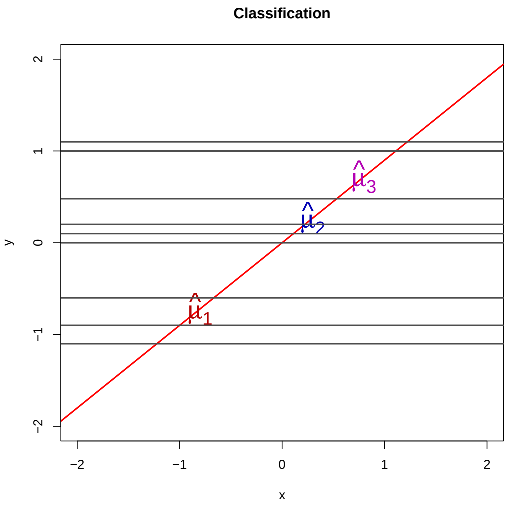
Hidden labels $z_{\text{test}:t}$ are iid uniform from S_{train}.
Generate $y_{\text{test}:t} \sim N(\beta x_{z_{\text{test}:t}}, \sigma_\epsilon^2)$



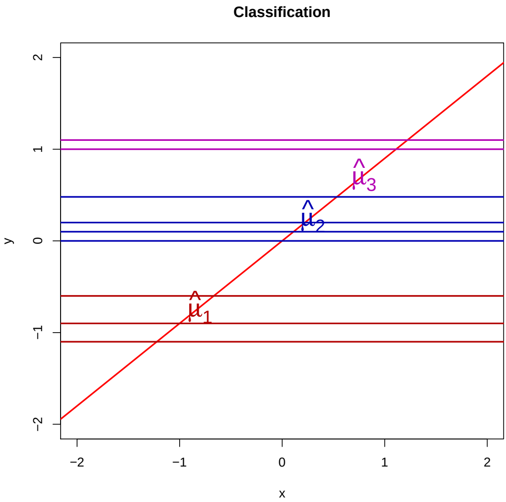
Classify $\hat{y}_{\text{test}:t}$



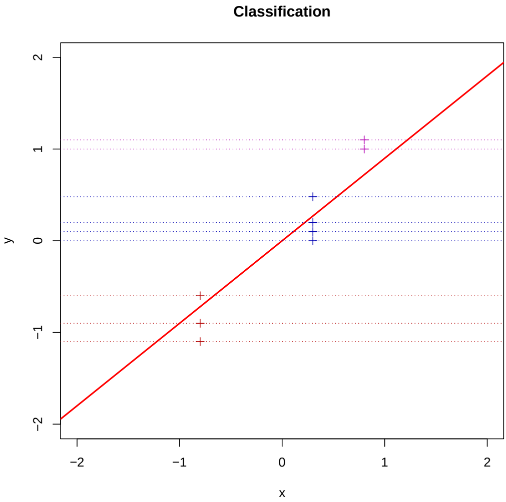
$$\hat{\mu}_{\text{test}:i} = \hat{\beta} x_{\text{test}:i}$$

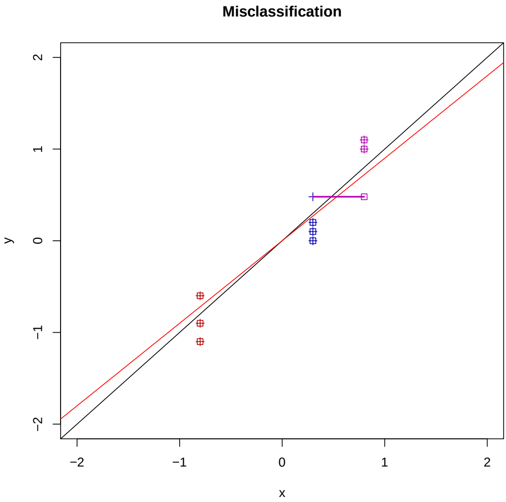


$\hat{z}_{\text{test}:t} = \text{argmin}_z \ell_{\hat{\mu}_z}(y_{\text{test}:t})$

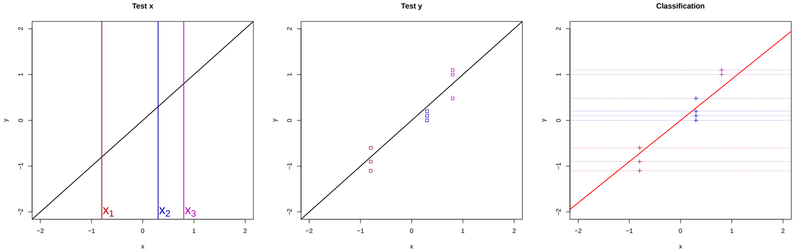


$$\hat{z}_{\text{test}:t} = \text{argmin}_z (\hat{\mu}_z - y_{\text{test}:t})^2$$


Classification
2
1
0
−1
−2
y
−2
−1
0
1
2
x


Misclassification
2
1
0
−1
−2
y
−2
−1
0
1
2
x

Toy example I



- 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 $y_{\text{test}:t} \sim N(\beta x_{z_{\text{test}:t}}, \sigma_\epsilon^2)$
- Classify $\hat{y}_{\text{test}:t}$ by maximum likelihood assuming $\hat{\beta}$ is correct. Thus:

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

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?

Toy example I: Analysis

- 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 $x_1, \ldots, x_{\ell-1}$.
- Let $\delta = \hat{\beta} - \beta$.

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

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

equivalently

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

where E_i is the event

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

with probability

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

Toy example I: Analysis

- Use the following conditioning

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

- An exact expression for expected misclassification is therefore
$$1 - \int_\epsilon \left[\int_x \left(1 - \left|\Phi\left(\frac{x}{\sigma_x}\right) - \Phi\left(\frac{x(\beta-\delta)+2\epsilon}{\sigma_x(\beta+\delta)}\right)\right|\right)^{\ell-1} d\Phi(\tfrac{x}{\sigma_x})\right] d\Phi(\tfrac{\epsilon}{\sigma_\epsilon})$$
- Question 1: Is this minimized at $\hat{\beta} = \beta$?

Answer: yes. (Part of a proof:)

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

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

In turn

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

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 symmetric 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 $\hat{\beta} = \beta$.

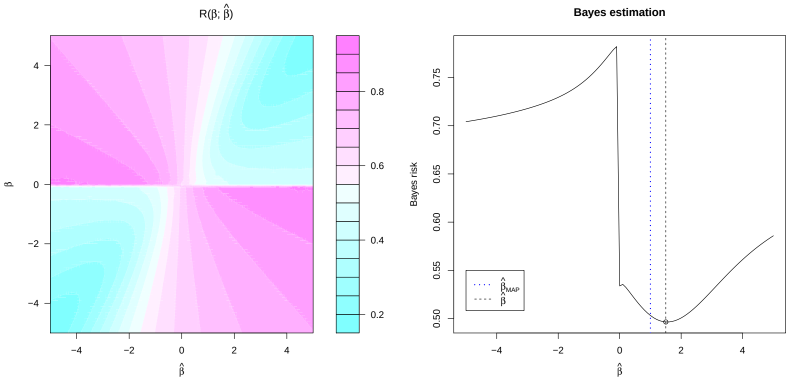
Toy example I: Estimation

- Second question: what about estimation?
- Take a Bayesian viewpoint: suppose we have a posterior distribution for $\hat{\beta}$, e.g. $\beta \sim N(\hat{\beta}_{MAP}, \sigma_\beta^2)$.
- For *least-squares regression*, we would use $\hat{\beta} = \hat{\beta}_{MAP}$, the posterior mean.
- For *identification*, we would choose

$$\hat{\beta}_{Bayes} = \text{argmin}_{\hat{\beta}} \int R(\beta; \hat{\beta}) \phi \left(\frac{\beta - \hat{\beta}_{MAP}}{\sigma_\beta} \right) d\beta$$

where R is the expected misclassification rate.

Toy example I: Estimation

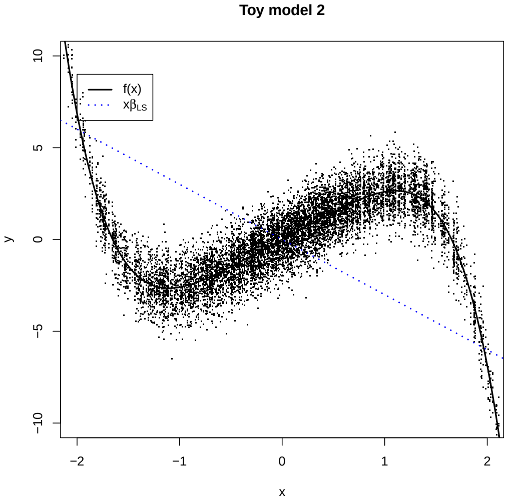


The Bayes point estimate for identification is larger than the Bayes point estimate for least-squares prediction.

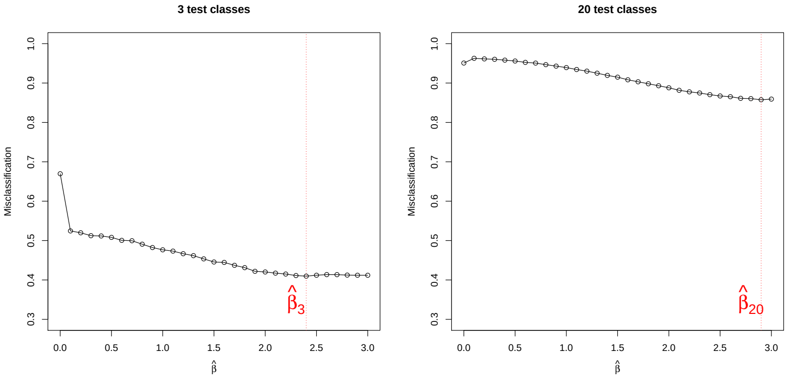
More questions

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 ℓ increases? What if ℓ increases while σ_ϵ^2 decreases?

Toy example IIa

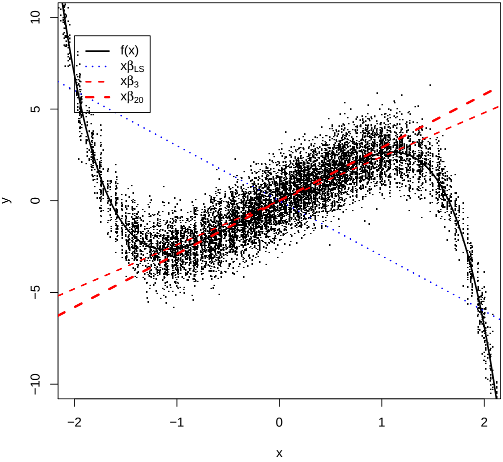


Toy example IIa



Effect of increasing ℓ.

Toy model 2



Why is this?

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

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

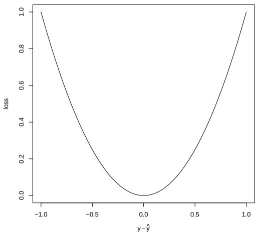
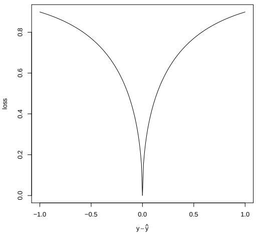
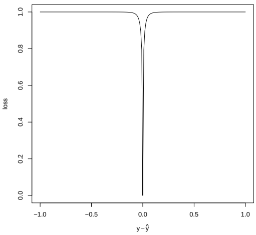
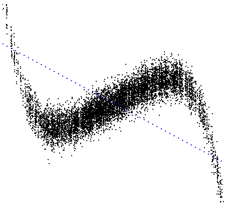
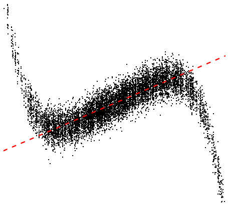
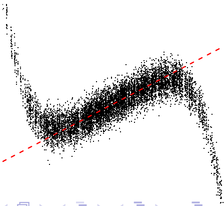
- Identification loss

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

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

Why is this?

Identification loss more closely resembles 0-1 loss as ℓ increases.

Squared error	$\ell = 3$	$\ell = 20$
		
		

Section 3

Methodology

Linear identification

Model fitting

- Inputs: features for training classes $\{x_{\text{train}:i}\}_{i=1}^{k}$ and points y_t with labels z_t for $t = 1, \ldots, T$. Features x have dimension p, responses y have dimension q.
- Outputs: $p \times q$ coefficient matrix B and $1 \times q$ intercept term b for a linear model
$$y \approx B^T x + b^T$$
and estimated covariance $\hat{\Sigma}_\epsilon$ for noise in y.

Identification

- Inputs: test class features $x_{\text{test}:\ i}$ for $i = 1, \ldots, \ell$. New point y_*.
- Output: label $\hat{z}_*$ given by
$$\hat{z}_* = \text{argmin}_{z=\text{test}:1,\ldots,\text{test}:\ell} d_{\hat{\Sigma}_\epsilon}(B^T x_z + b, y_*)^2$$
where $d_\Sigma(\cdot, \cdot)$ is the Mahalanobis distance.
- Evaluation: misclassification comparing $\hat{z}_*$ with true label z_*.

Model fitting

- Inputs: features for training classes $\{x_{\text{train}:i}\}_{i=1}^k$ and points y_t with labels z_t for $t = 1, \ldots, T$.

Procedure

1. Estimate $\hat{\Sigma}_x$ from sample covariance of $\{x_{\text{train}:i}\}_{i=1}^k$ and $\hat{\mu}_x$ from sample mean. Let $\hat{P}_x$ be the distribution of $N(\hat{\mu}_x, \hat{\Sigma}_x)$
2. Estimate $\hat{\Sigma}_\epsilon$ from pooled sample within-class covariance of y_t
3. Maximize for B, b:

$$\sum_{t=1}^{T} \left[\int_{\mathbb{R}^p} I\{d(B^T x + b^T, y_t) < d(B^T x_{z_t} + b^T, y_t)\} d\hat{P}_x(x) \right]^{\ell-1}$$

4. Output B, b, $\hat{\Sigma}_\epsilon$

Computation

- Maximize for B, b:

$$\sum_{t=1}^{T} 1 - \mathcal{L}((x_{z_t}, y_t); B, b)$$

where

$$\mathcal{L}((x_{z_t}, y_t), B, b) = 1 - \left[\int_{\mathbb{R}^p} I\{d(B^T x + b^T, y_t) < d(B^T x_{z_t} + b^T, y_t)\} d\hat{F}$$

- Use iteratively reweighted least squares. In iteration $k+1$, update

$$(B^{(k+1)}, b^{(k+1)}) = \text{argmin}_{B,b} \sum_{t=1}^{T} w_t^{(k)} ||y_t - B^T x_{z_t} - b^T||^2$$

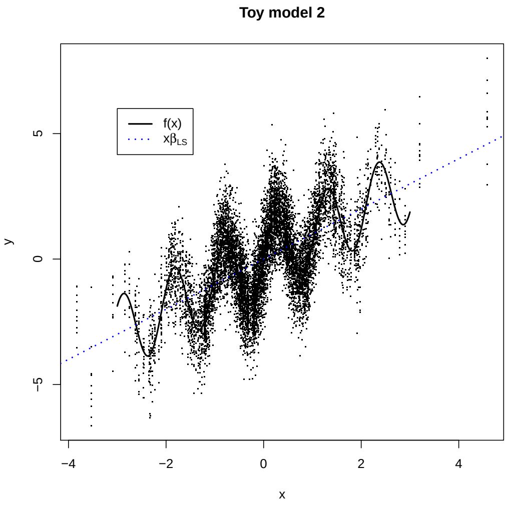
where

$$w^{(k)} = \frac{\mathcal{L}((x_{z_t}, y_t), B^{(k)}, b^{(k)})}{||y_t - (B^{(k)})^T x_{z_t} - (b^{(k)})^T||^2}$$

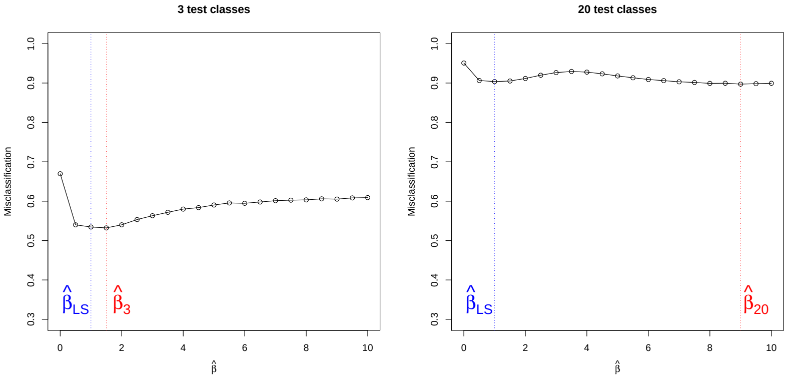
Section 4

Issues

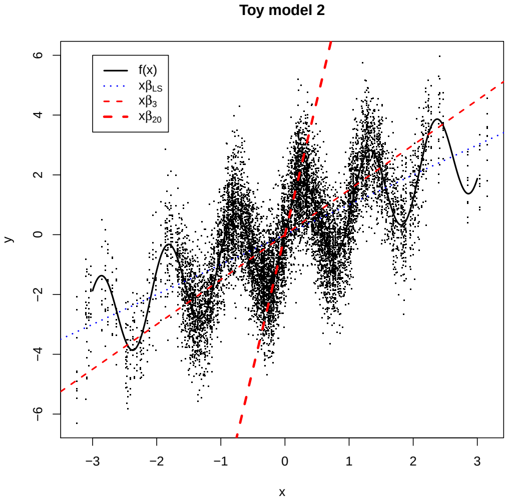
Toy example IIb



Toy example IIb



Effect of increasing ℓ.



Effect of increasing ℓ: 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

Solution: Label sets

- One option is to only use small ℓ. However, this is not satisfactory since with good signal-to-noise ratio, we should be able to identify a stimuli from a large set of candidates.
- Develop a method for producing a *set of labels* for each point rather than a single label. Evaluate the method using a metric such as precision-recall.
- The labeller would assign a proportional number of labels to each point as ℓ increases, thus maintaining coverage probability. Thus, it will no longer become optimal to just "give up" on global estimation as ℓ increases.
- It would be desirable to find a loss function so that the optimal parametric model is fixed as ℓ varies.

References

- Kay, KN., Naselaris, T., Prenger, R. J., and Gallant, J. L. "Identifying natural images from human brain activity". *Nature* (2008)
- 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)