

# Supervised Evaluation of Representations

Charles Zheng

Stanford University

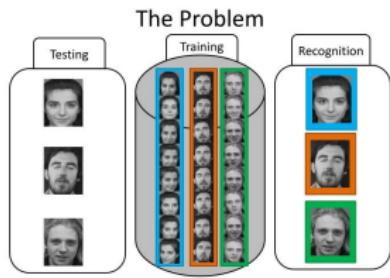
May 1, 2017

## Section 1

### Randomized Classification

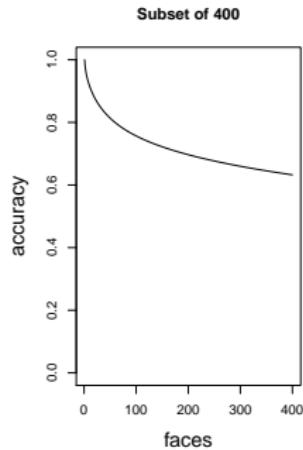
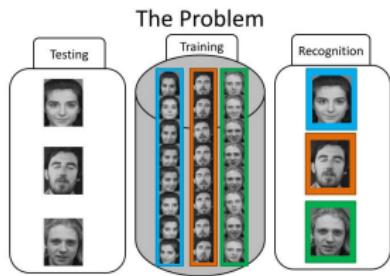
# Facial recognition problem

Let's say I have a database of users,  $i = 1, \dots, k$ , with photos  $\vec{z}_j$  that are labelled.



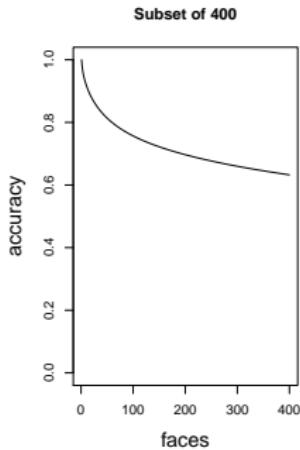
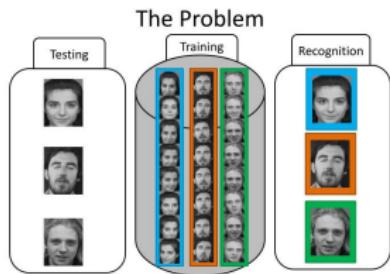
# Facial recognition problem

Let's say I have a database of users,  $i = 1, \dots, k$ , with photos  $\vec{z}_i$  that are labelled.



# Facial recognition problem

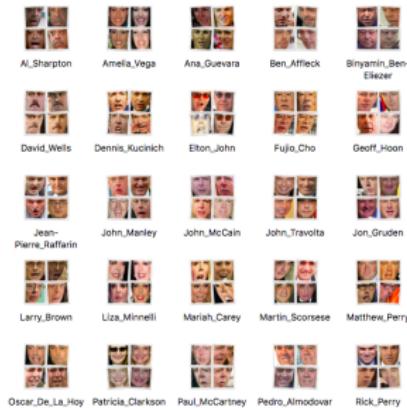
Let's say I have a database of users,  $i = 1, \dots, k$ , with photos  $\vec{z}_i$  that are labelled.



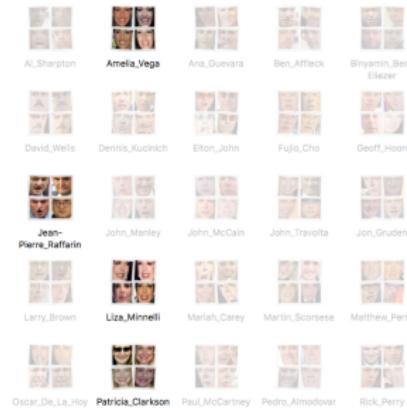
How does the accuracy scale with the number of classes (faces)?

# Setup

1. Population of categories  $\pi(y)$

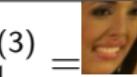
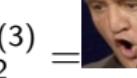
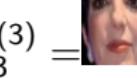
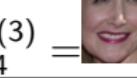


2. Subsample  $k$  labels,  $y_1, \dots, y_k$



# Setup

3. Collect training and test data  $x_j^{(i)}$  (faces) for labels (people)  $\{y_1, \dots, y_k\}$ .

Label	Training			Test
$y_1 = \text{Amelia}$	$x_1^{(1)} =$ 	$x_1^{(2)} =$ 	$x_1^{(3)} =$ 	$x_1^* =$ 
$y_2 = \text{Jean-Pierre}$	$x_2^{(1)} =$ 	$x_2^{(2)} =$ 	$x_2^{(3)} =$ 	$x_2^* =$ 
$y_3 = \text{Liza}$	$x_3^{(1)} =$ 	$x_3^{(2)} =$ 	$x_3^{(3)} =$ 	$x_3^* =$ 
$y_4 = \text{Patricia}$	$x_4^{(1)} =$ 	$x_4^{(2)} =$ 	$x_4^{(3)} =$ 	$x_4^* =$ 

4. Train a classifier and compute test error.

# Setup

3. Collect training and test data  $x_j^{(i)}$  (faces) for labels (people)  $\{y_1, \dots, y_k\}$ .

Label	Training			Test
$y_1 = \text{Amelia}$	$x_1^{(1)} =$ 	$x_1^{(2)} =$ 	$x_1^{(3)} =$ 	$x_1^* =$ 
$y_2 = \text{Jean-Pierre}$	$x_2^{(1)} =$ 	$x_2^{(2)} =$ 	$x_2^{(3)} =$ 	$x_2^* =$ 
$y_3 = \text{Liza}$	$x_3^{(1)} =$ 	$x_3^{(2)} =$ 	$x_3^{(3)} =$ 	$x_3^* =$ 
$y_4 = \text{Patricia}$	$x_4^{(1)} =$ 	$x_4^{(2)} =$ 	$x_4^{(3)} =$ 	$x_4^* =$ 

4. Train a classifier and compute test error.

**Can we analyze how error depends on  $k$ ?**

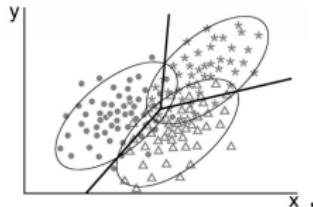
## Key assumption: marginal classifier

- The classifier is *marginal* if it learns a model *independently* for each class.

# Key assumption: marginal classifier

- The classifier is *marginal* if it learns a model *independently* for each class.

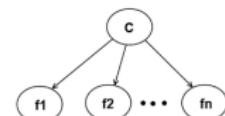
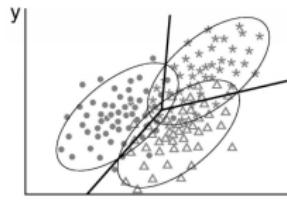
- Examples: LDA/QDA



# Key assumption: marginal classifier

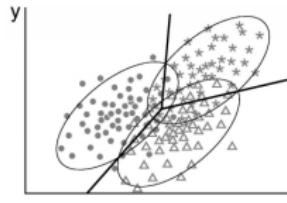
- The classifier is *marginal* if it learns a model *independently* for each class.

- Examples: LDA/QDA

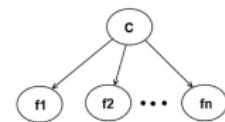


# Key assumption: marginal classifier

- The classifier is *marginal* if it learns a model *independently* for each class.



- Examples: LDA/QDA , naïve Bayes
- Non-marginal classifiers: Multinomial logistic, multilayer neural networks, k-nearest neighbors

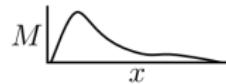


# Definitions

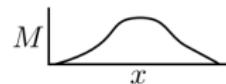
$\hat{F}_{y^{(i)}}$  is the empirical distribution obtained from the training data for label  $y^{(i)}$ .

Classification Rule

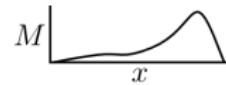
$$M_{y^{(1)}}(x) = \mathcal{M}(\hat{F}_{y^{(1)}})(x)$$



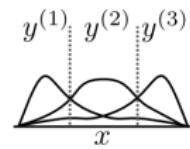
$$M_{y^{(2)}}(x) = \mathcal{M}(\hat{F}_{y^{(2)}})(x)$$



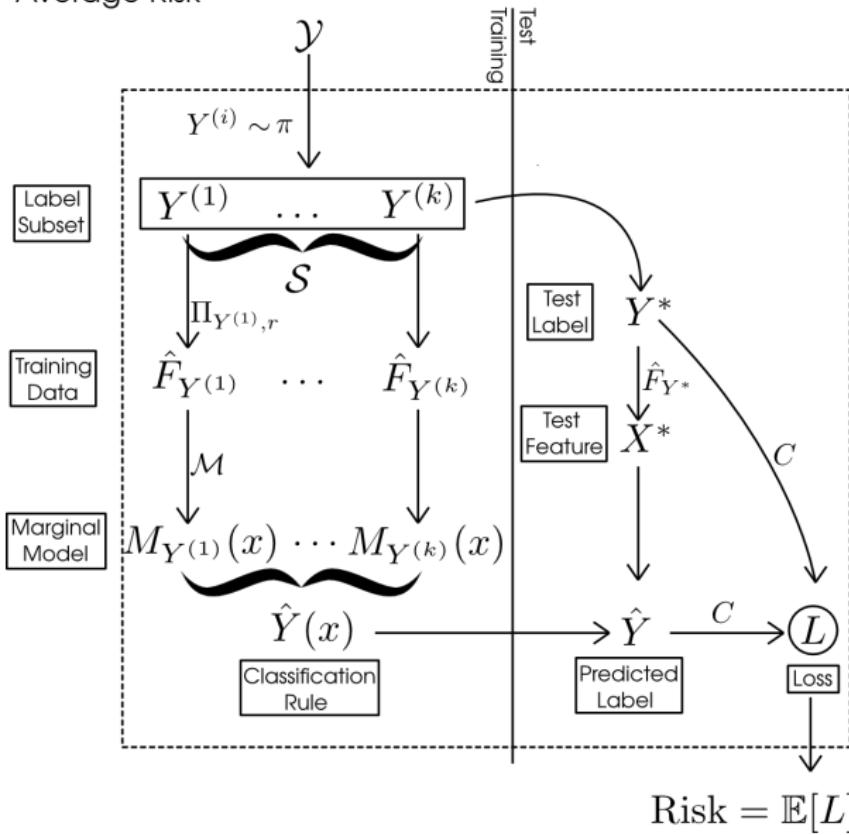
$$M_{y^{(3)}}(x) = \mathcal{M}(\hat{F}_{y^{(3)}})(x)$$



$$\hat{Y}(x) = \operatorname{argmax}_{y \in \mathcal{S}} M_y(x)$$



## Average Risk



# Theoretical Result

**Theorem.** (Z., Achanta, Benjamini.) Suppose  $\pi$ ,  $\{F_y\}_{y \in \mathcal{Y}}$  and marginal classifier  $\mathcal{F}$  satisfy (*some regularity condition*). Then, there exists some function  $\bar{D}(u)$  on  $[0, 1] \rightarrow [0, 1]$  such that the  $k$ -class average risk is given by

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$

# Theoretical Result

**Theorem.** (Z., Achanta, Benjamini.) Suppose  $\pi$ ,  $\{F_y\}_{y \in \mathcal{Y}}$  and marginal classifier  $\mathcal{F}$  satisfy (*some regularity condition*). Then, there exists some function  $\bar{D}(u)$  on  $[0, 1] \rightarrow [0, 1]$  such that the  $k$ -class average risk is given by

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$

What is this  $\bar{D}(u)$  function? We will explain in the following toy example...

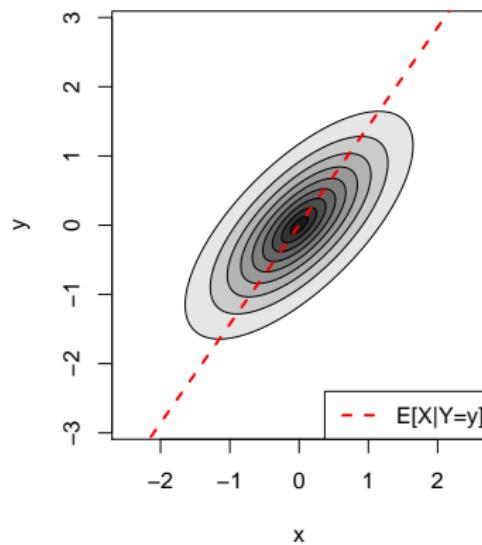
## Toy example

$$Y_1, \dots, Y_k \stackrel{iid}{\sim} N(0, 1);$$

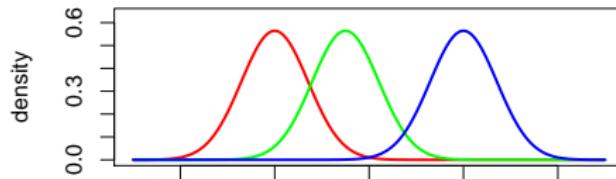
# Toy example

$Y_1, \dots, Y_k \stackrel{iid}{\sim} N(0, 1);$

$X|Y \sim N(\rho Y, 1 - \rho^2)$  i.e.  $(Y, X) \sim N(0, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix})$ .

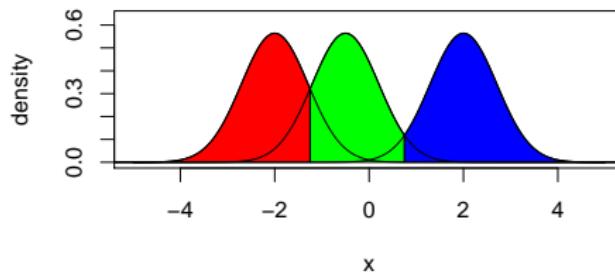


# Toy example



- Suppose  $k = 3$ , and we draw  $Y_1, Y_2, Y_3$ .
- The *Bayes rule* is the optimal classifier and depends on knowing the true densities:
$$\hat{y}(x) = \operatorname{argmax}_{y_i} p(x|y_i)$$
- The *Bayes Risk*, which is the misclassification rate of the optimal classifier.

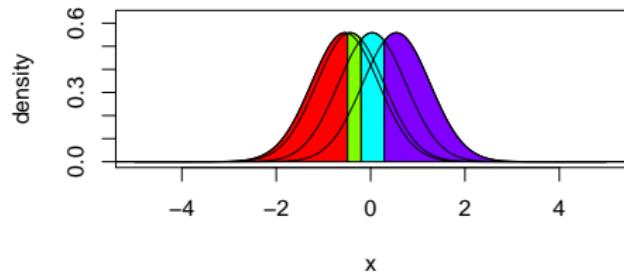
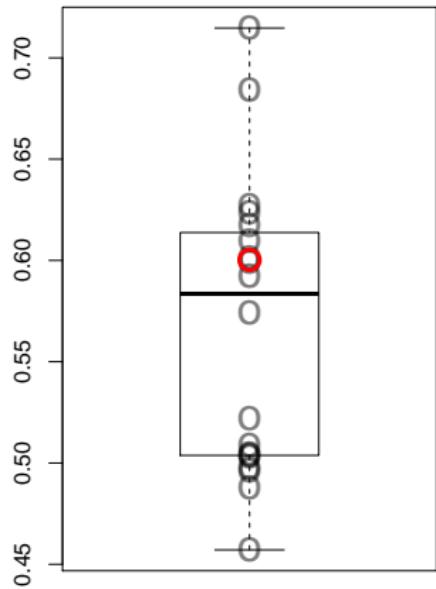
# Toy example



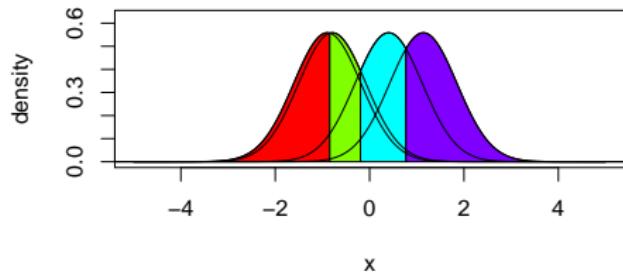
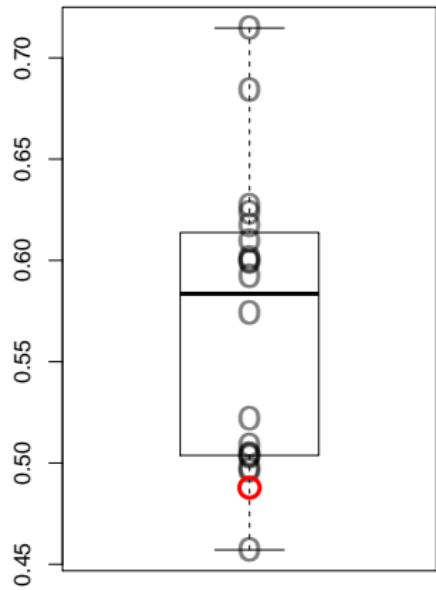
- The *Bayes Risk* is the expected test error of the Bayes rule,

$$\frac{1}{k} \sum_{i=1}^k \Pr[\hat{y}(x) \neq Y | Y = y_i]$$

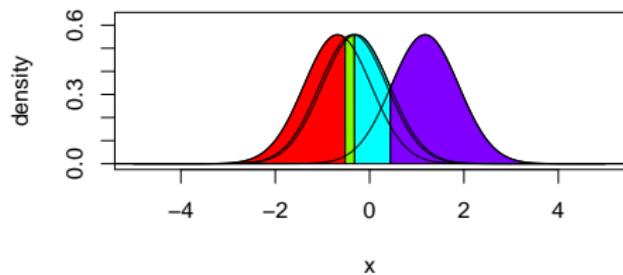
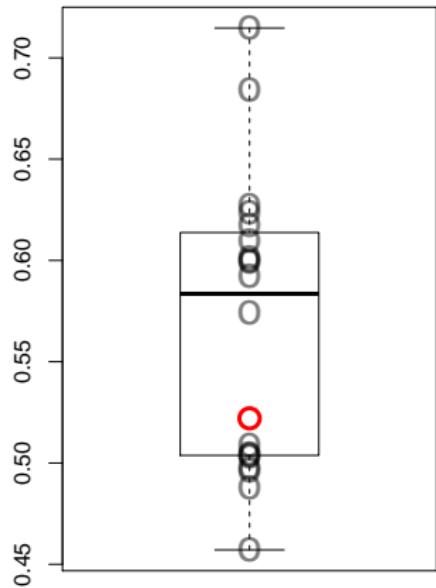
# Toy example



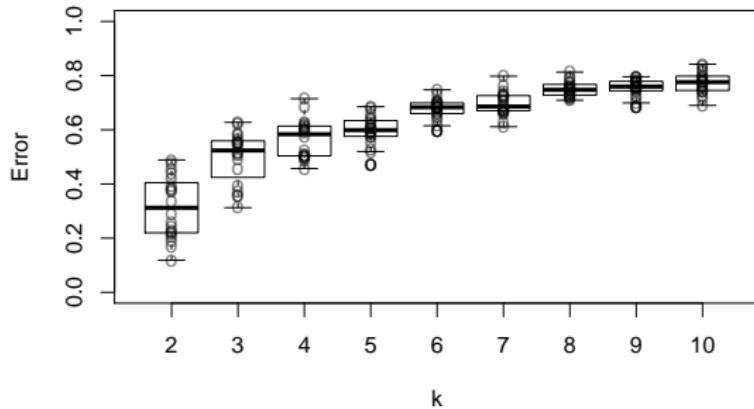
# Toy example



# Toy example

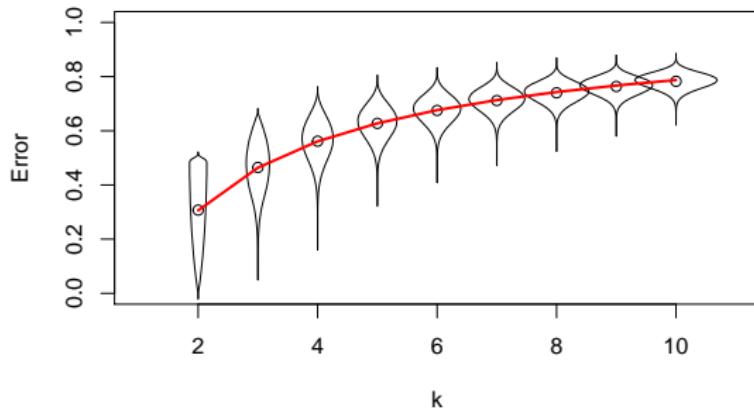


# Toy example



# Toy example

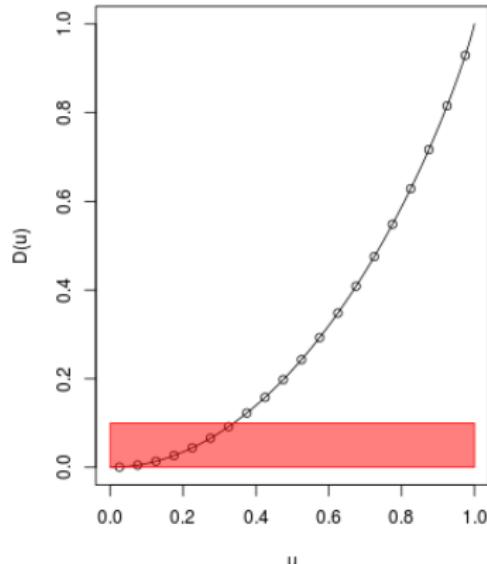
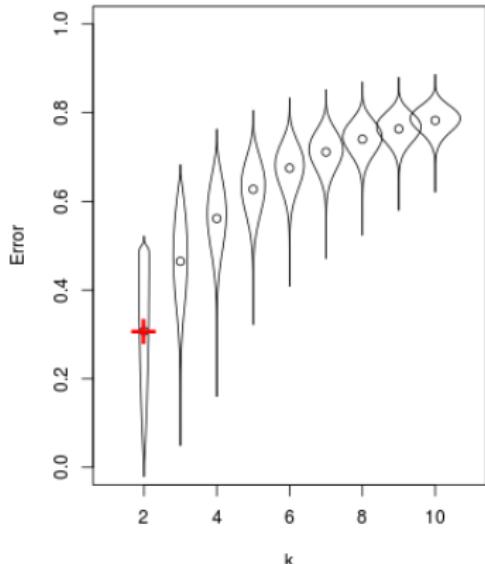
$\rho = 0.7$



# Computing average risk

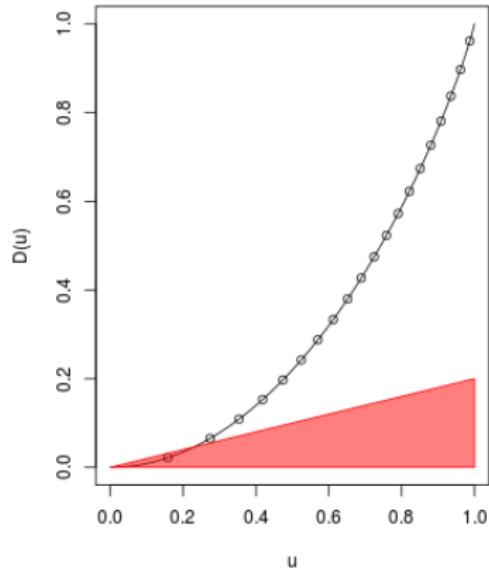
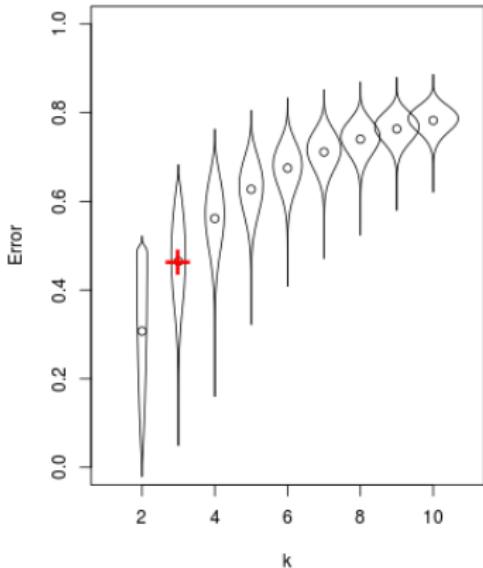
$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$

$(k = 2)$



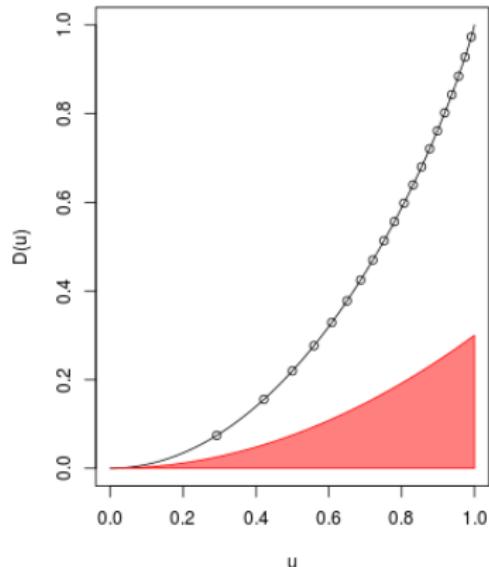
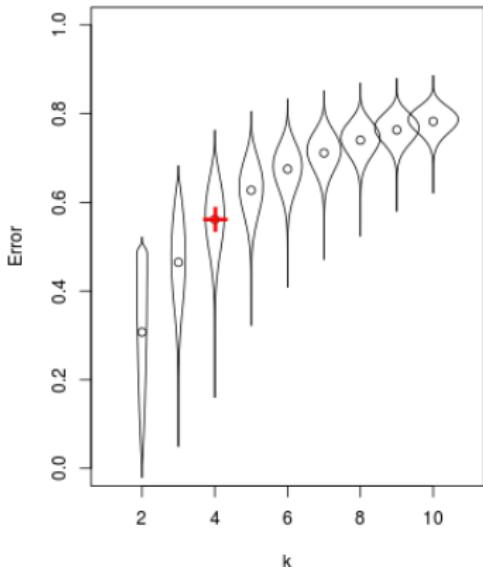
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



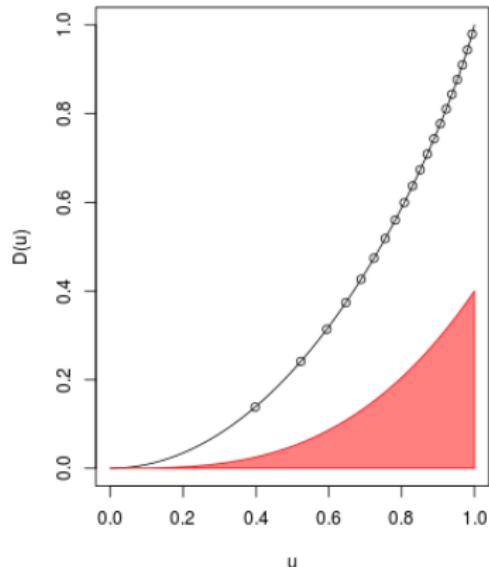
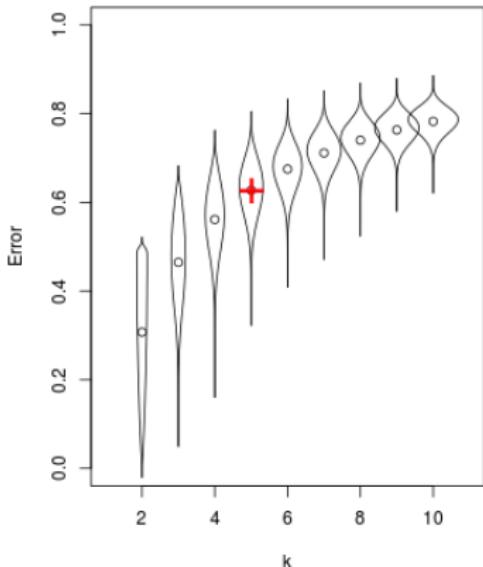
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



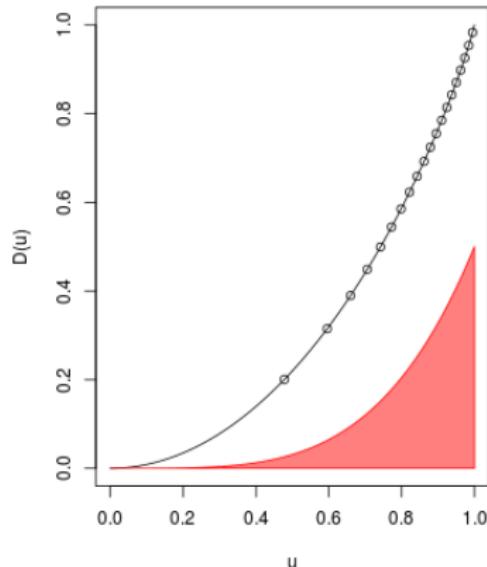
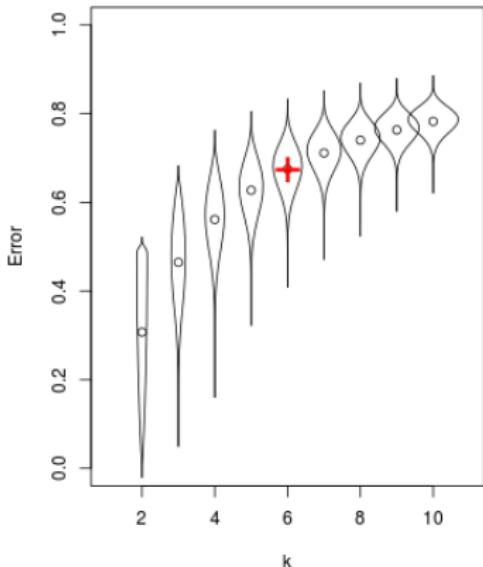
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



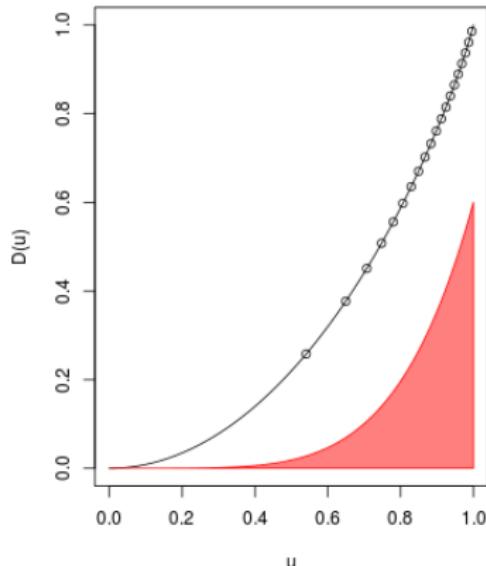
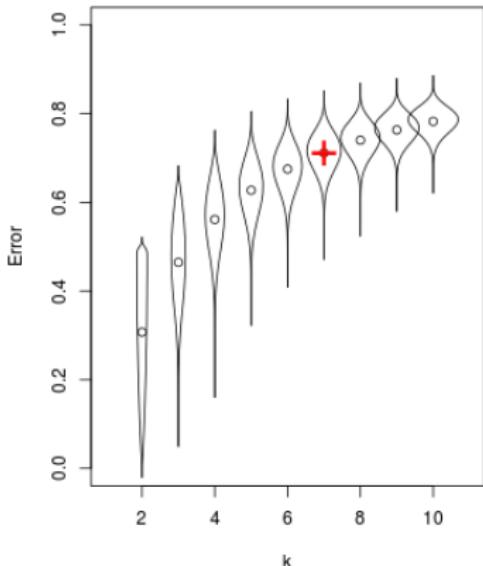
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



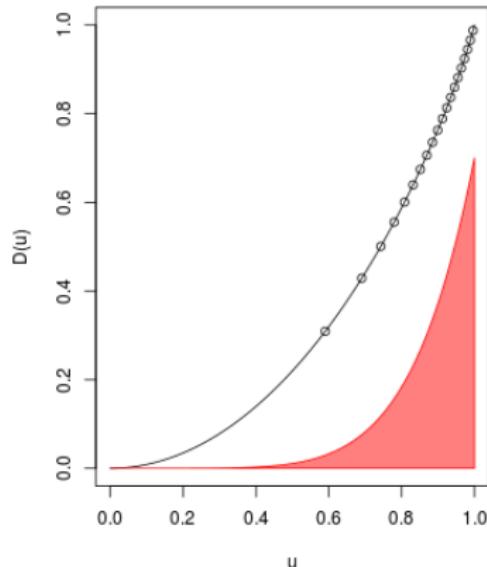
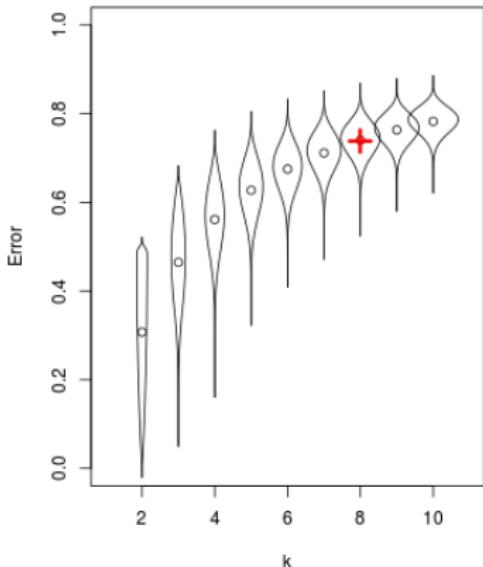
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



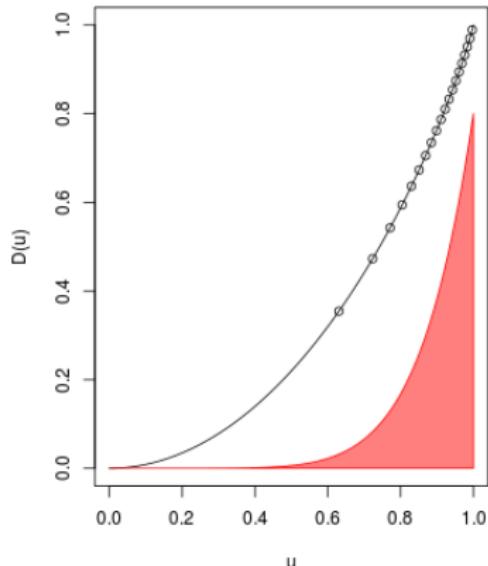
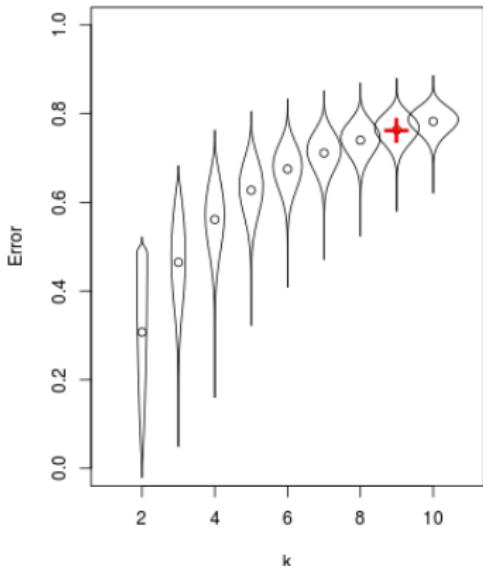
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



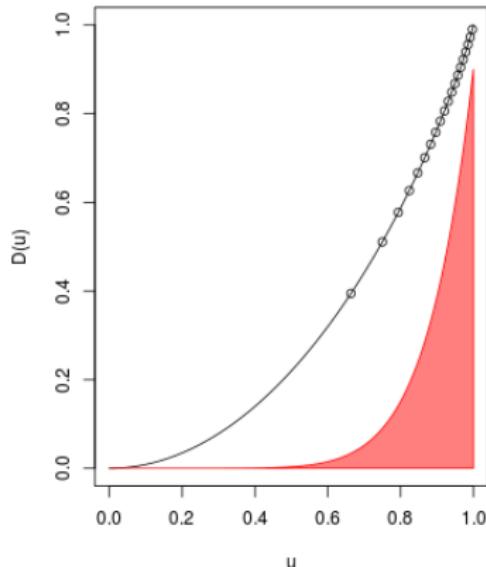
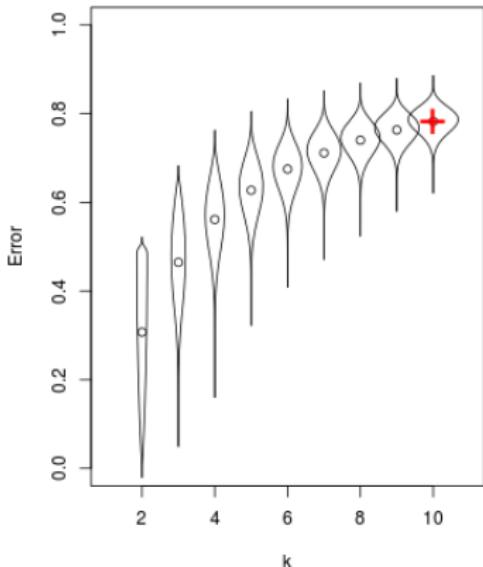
# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$



# Computing average risk

$$\text{AvRisk}_k = (k - 1) \int \bar{D}(u) u^{k-2} du.$$

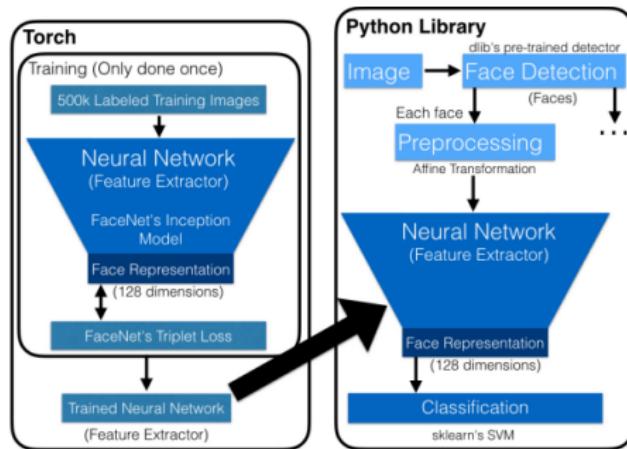


## Implication: estimate $\bar{D}(u)$ to predict risk

- Theoretical result links  $k$ -class average risk to  $\bar{D}(u)$  function
- In real data, we do not know  $\bar{D}(u)$  since it depends on the unknown joint distribution
- However, given a model, we can estimate  $\bar{D}(u)$

# Facial recognition example

- Data: faces from “Labeled Faces in the Wild.”
- 1672 people with at least 2 photos
- Featurization: trained neural network from OpenFace

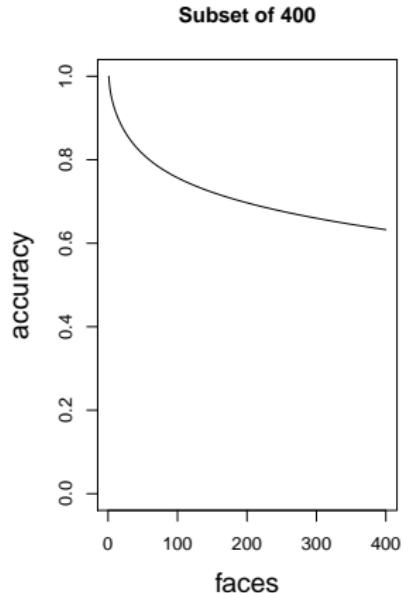


# Facial recognition example

- Let us first subsample 400 faces (out of 1672)
- Randomly choose 1 face as training and 1 as test for each person
- Use 1-nearest neighbor.
  - NOTE: 1-NN with 1 example/class is equivalent to LDA with  $\Sigma = I$ : this fits marginal classifier assumption!

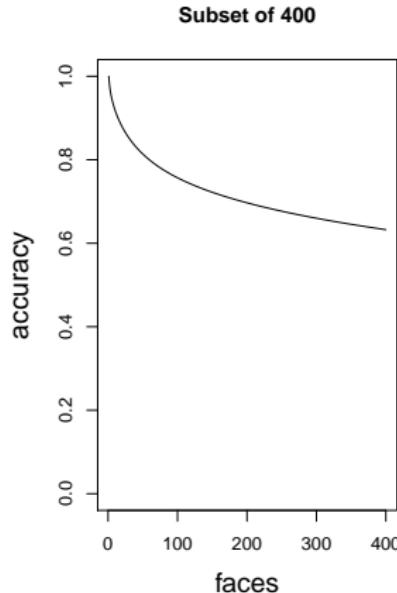
# Facial recognition example

- Let us first subsample 400 faces (out of 1672)
- Randomly choose 1 face as training and 1 as test for each person
- Use 1-nearest neighbor.
  - NOTE: 1-NN with 1 example/class is equivalent to LDA with  $\Sigma = I$ : this fits marginal classifier assumption!



# Facial recognition example

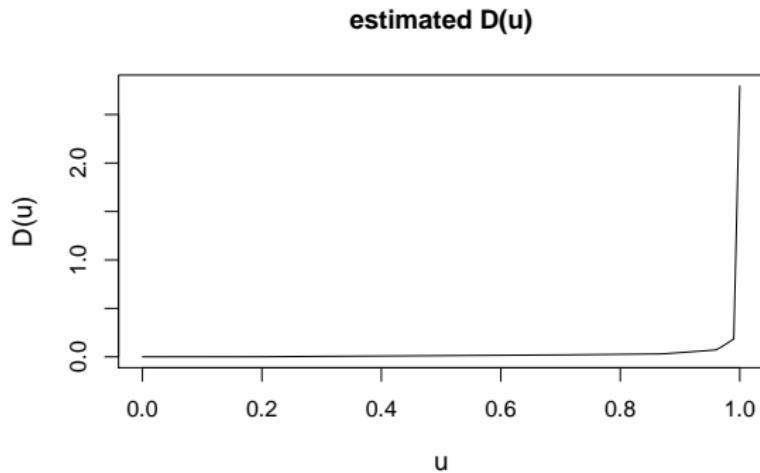
- Let us first subsample 400 faces (out of 1672)
- Randomly choose 1 face as training and 1 as test for each person
- Use 1-nearest neighbor.
  - NOTE: 1-NN with 1 example/class is equivalent to LDA with  $\Sigma = I$ : this fits marginal classifier assumption!



Can we predict the accuracy on the full set of 1672?

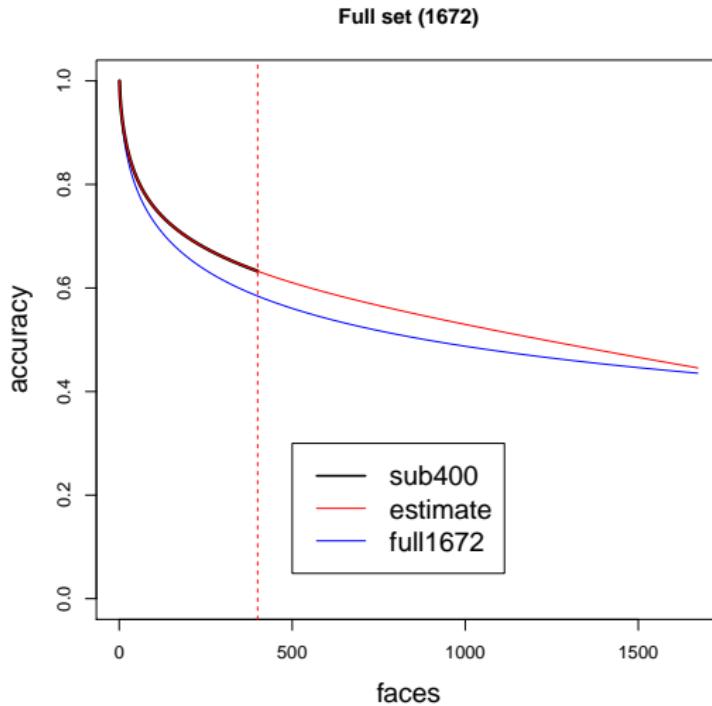
# Estimated $\bar{D}(u)$

Using linear spline basis ( $p = 10000$ ) and nonnegativity constraint.

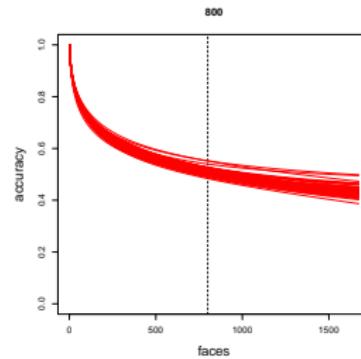
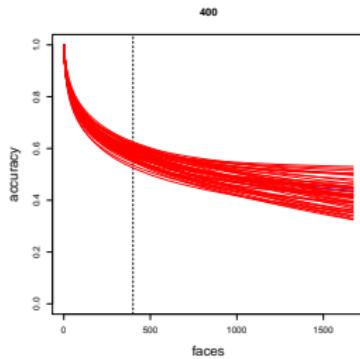
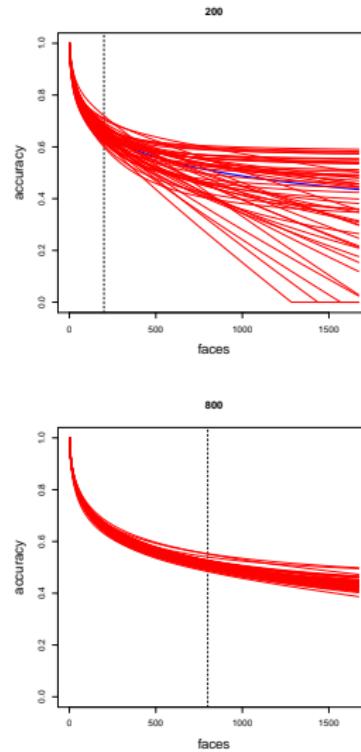
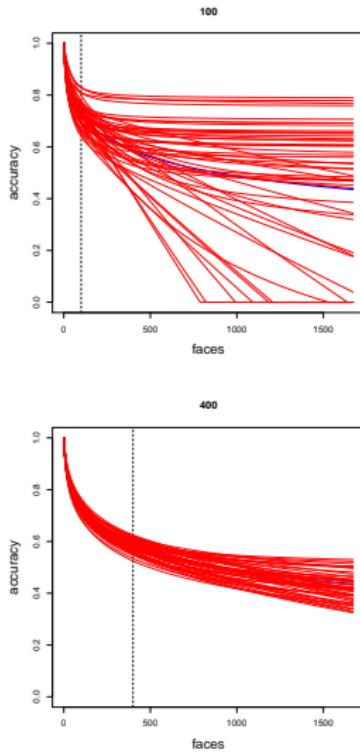


# Estimated risk

Compare to test risk at  $K = 1672$

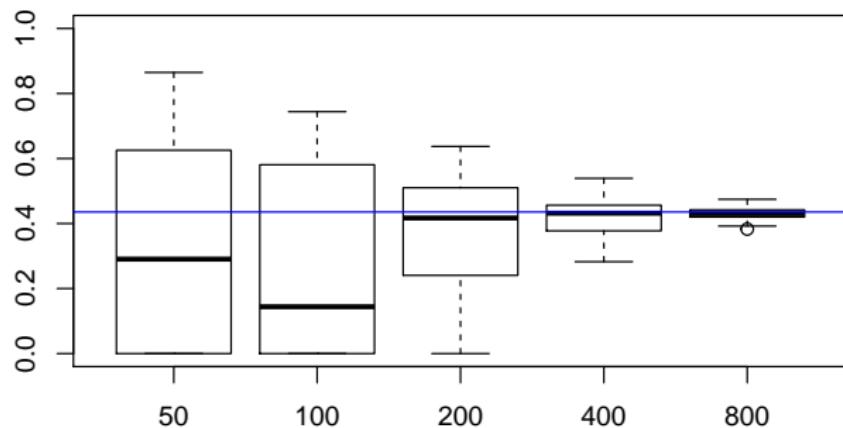


# Estimated risk: more experiments



# Estimated risk: more experiments

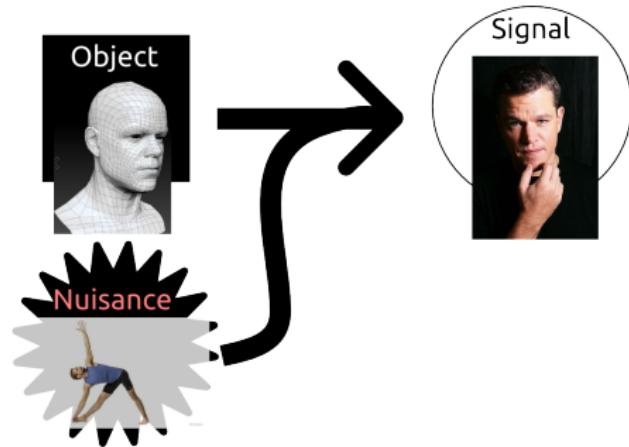
Predicted accuracy (1672)



## Section 2

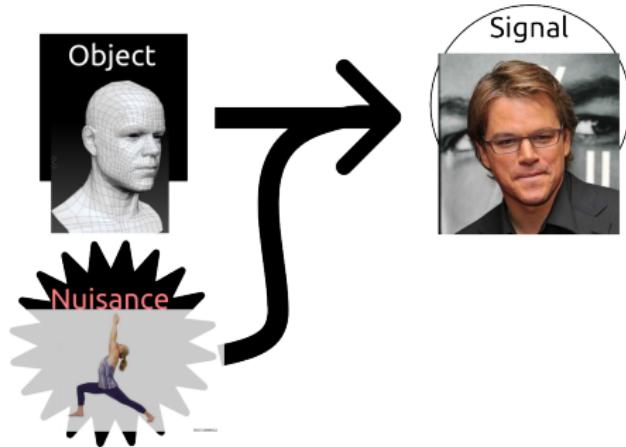
### Geometry of representations

# Example: face recognition



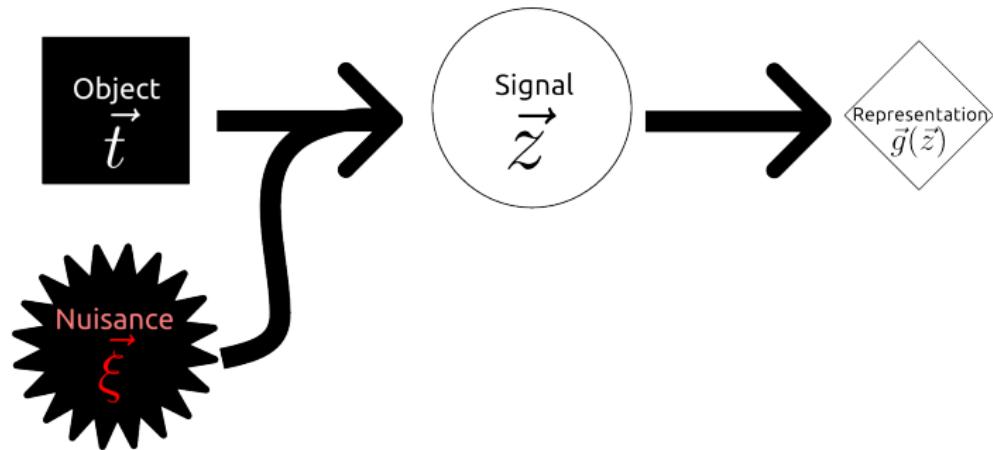
In face recognition, the *pose* (including hairstyle) and *lighting* are nuisance parameters.

# Example: face recognition



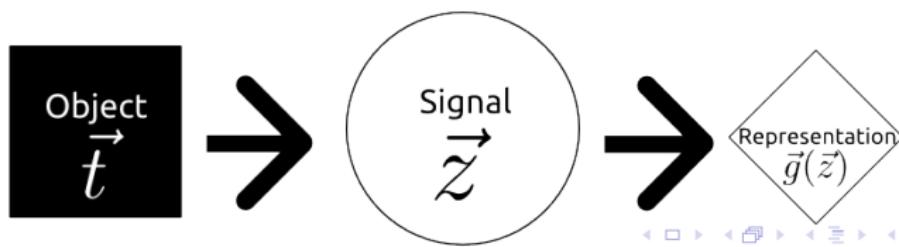
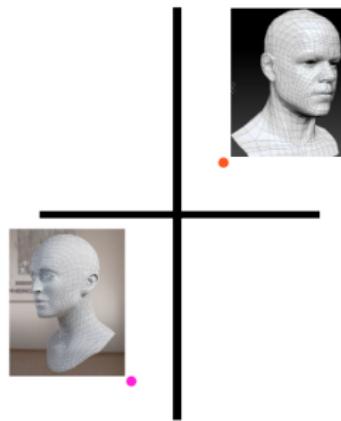
The same object can map to multiple signals.

# What is a representation?

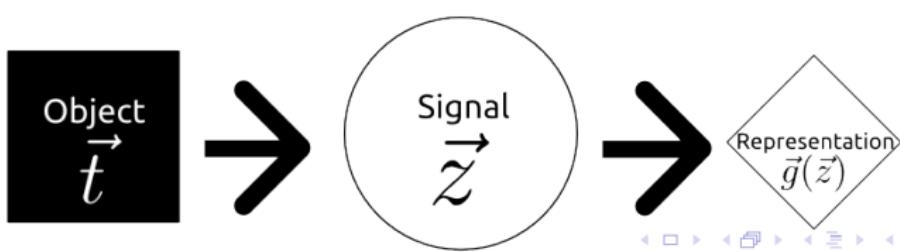
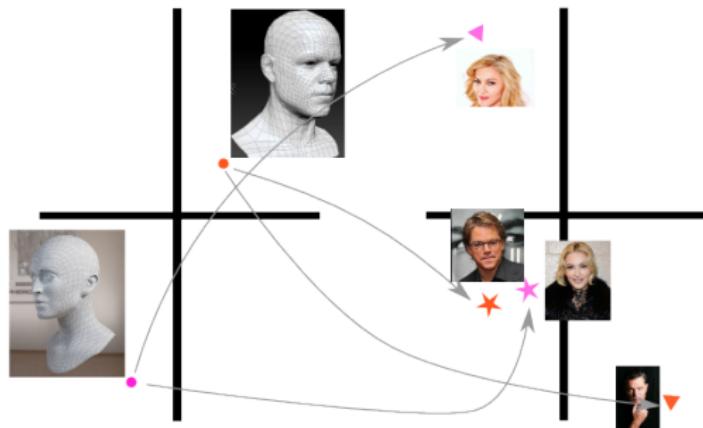


A dimensionality-reducing mapping  $\vec{g}$  of the signal.

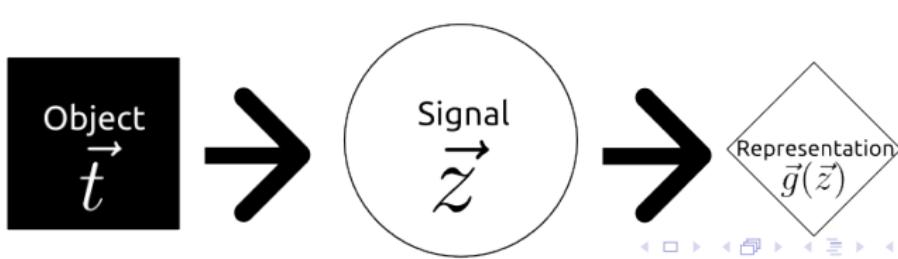
# A good representation...



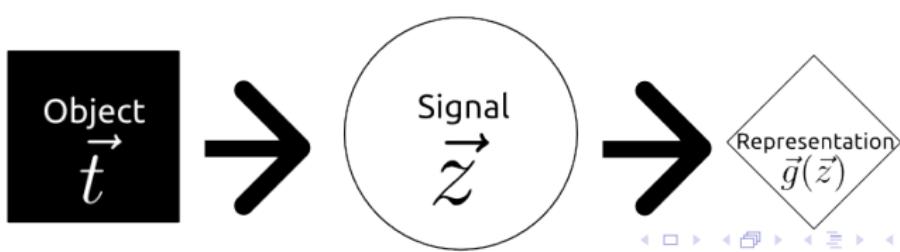
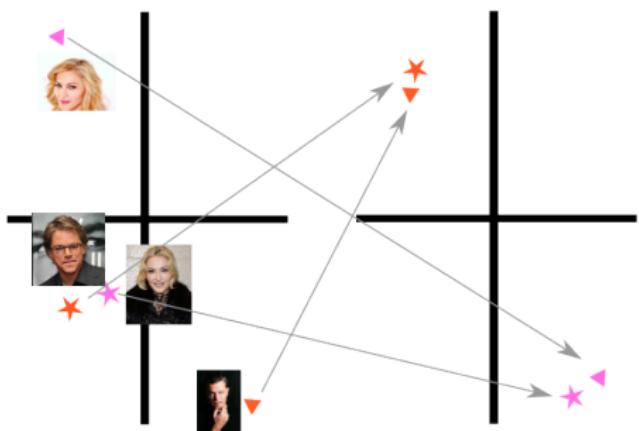
# A good representation...



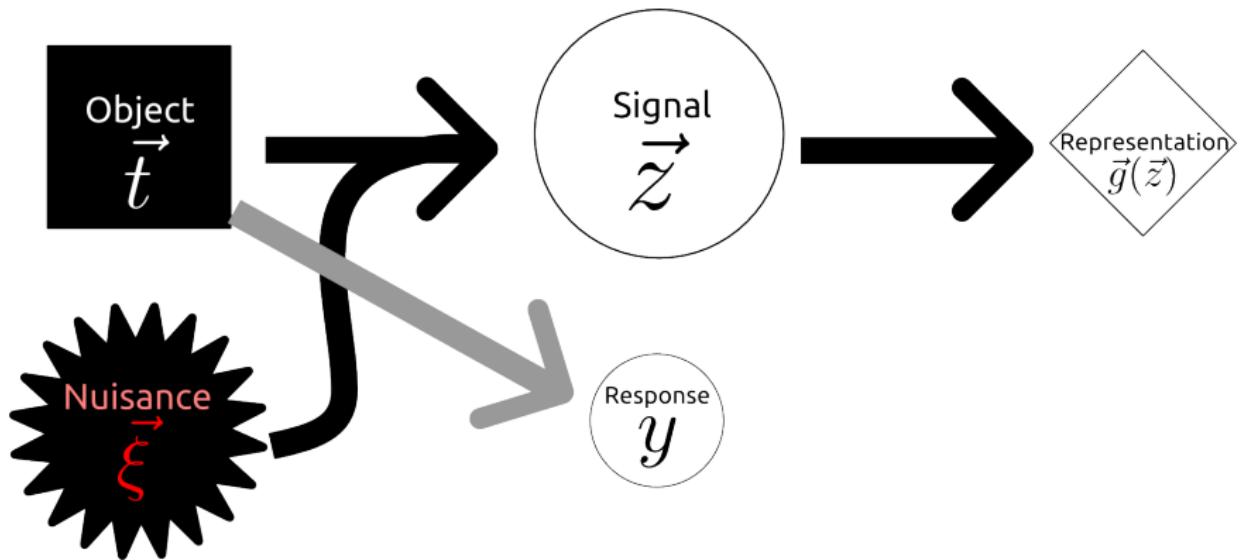
# A good representation...



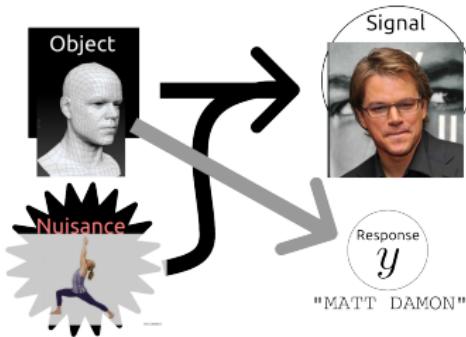
...captures the object space geometry



# Supervised evaluation of representations

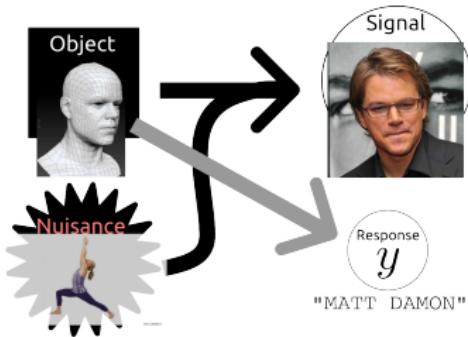


# Example: face recognition



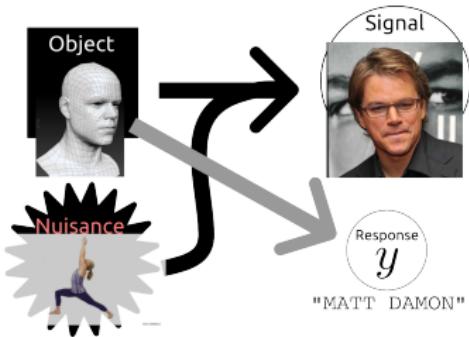
- The ID of the individual is an appropriate *response* variable...

# Example: face recognition



- The ID of the individual is an appropriate *response* variable...
- ...because two photos labeled with the same ID must belong to the same object  $\vec{t}$

## Example: face recognition

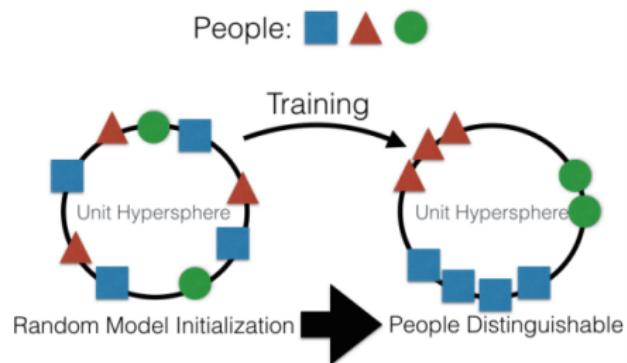


- The ID of the individual is an appropriate *response* variable...
- ...because two photos labeled with the same ID must belong to the same object  $\vec{t}$
- That is, for  $d(y, y')$  being the zero-one distance,

$$d(y, y') = 0 \Leftrightarrow d(\vec{t}, \vec{t}') = 0.$$

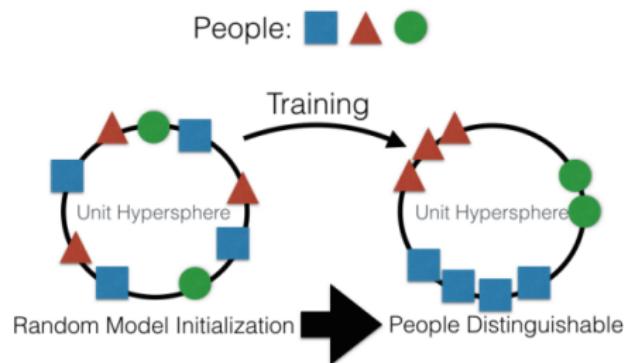
# Methods for supervised evaluation of representations

- *Triplet loss* (Schroff 2015)



# Methods for supervised evaluation of representations

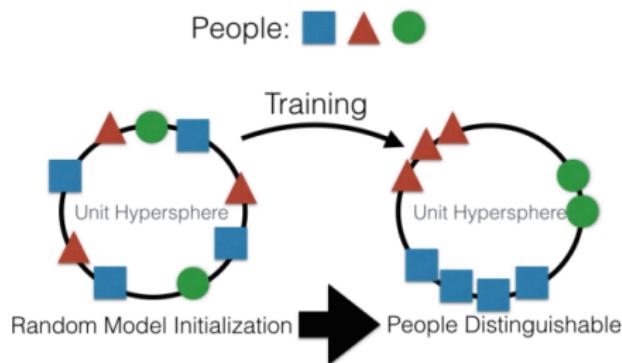
- *Triplet loss* (Schroff 2015)



- Average risk of randomized classification (using e.g. 1-nearest neighbor)

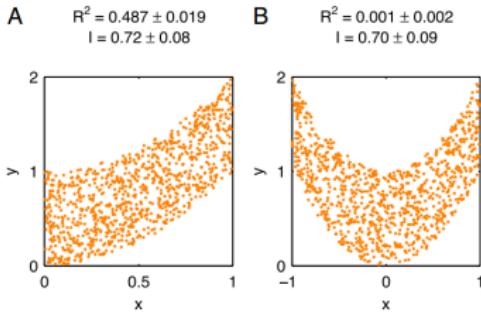
# Methods for supervised evaluation of representations

- *Triplet loss* (Schroff 2015)



- Average risk of randomized classification (using e.g. 1-nearest neighbor)
- *Mutual information*  $I(g(\vec{Z}), \vec{Y})$

# Mutual information $I(X; Y)$



Introduced in Shannon's 1948 paper, "A mathematical theory of communication"

$$I(X; Y) = \int \log \left( \frac{p(x, y)}{p(x)p(y)} \right) p(x, y) dx dy$$

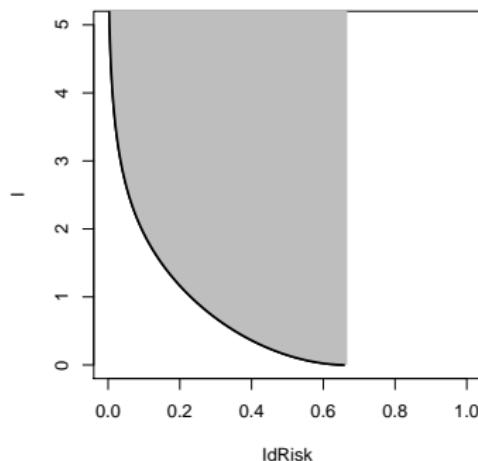
Image credit Kinney et al. 2014.

# Result 1. Lower bound for mutual information

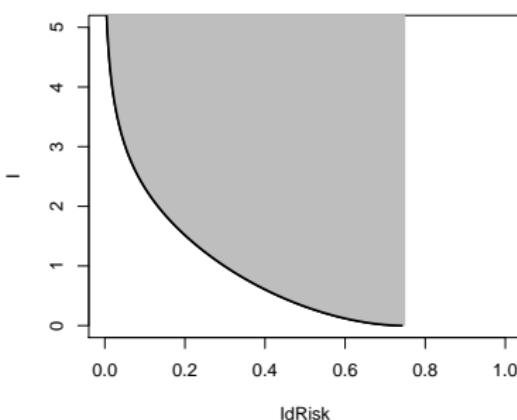
- **Theorem.** (Z., Benjamini 2017) There exists a function  $h_k$  such that

$$I(\vec{g}(\vec{Z}); \vec{Y}) \geq h_k(\text{AvRisk}_k).$$

$h_3$



$h_4$



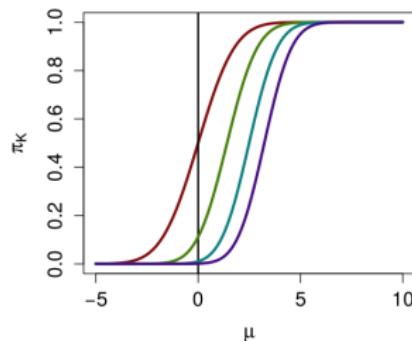
## Result 2. Limiting behavior of accuracy curves

(Z., Benjamini 2016) Define  $\text{ABA}_k$  as the Bayes identification accuracy (or average Bayes classification accuracy). Then under a particular high-dimensional limit,

$$\text{ABA}_k \approx \pi_k(\sqrt{2I(X; Y)}) \quad (1)$$

The function  $\pi_k$  is given by

$$\pi_k(c) = \int_{\mathbb{R}} \phi(z - c) \Phi(z)^{k-1} dz.$$



Legend:  $K = \{ \boxed{2}, \boxed{9}, \boxed{99}, \boxed{999} \}$

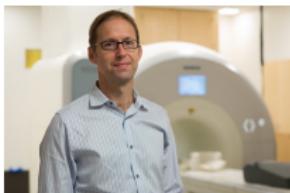
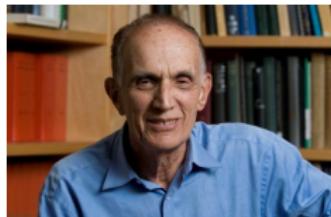
## Section 3

### Acknowledgements

# Co-advisors



# Committee



# Collaborators



# ¡Compadres!



## Section 4

The end