# APPLIED MACHINE LEARNING

***Interactive Lecture***

Classification with
Gaussian Mixture Models (GMM) + Bayes
K-nearest neighbors

## Happy Halloween


SWISS ROBOTICS DAY
SRD 2024 EDITION
ORGANIZING COMMITTEE
PREVIOUS EDITIONS
CONTACT
SWISS ROBOTICS DAY 2024
1st of November • Messe und Congress Center Basel
Messepl. 21, 4058 Basel
15 years at the forefront of the Swiss robotics exhibitions & networking scene
2 : 17 : 22
Days
Hours
Minutes
Click for Details & Registration
Messe und Congress Center
BASEL
1 NOVEMBER 2024

# ELLIS PHD PROGRAM

https://ellis.eu/news/ellis-phd-program-call-for-applications-2024

ELLIS Society

News Events Research People Sponsorship About Contact

## ELLIS PhD Program: Call for applications 2024

Apply by November 15, 2024, to join the ELLIS PhD Program in 2025

02 October 2024 News

ELLISPhD


*The call is open*

The **ELLIS PhD program** is a key pillar of the ELLIS initiative whose goal is to foster and educate the best talent in machine learning and related research areas by pairing outstanding students with leading academic and industrial researchers in Europe. The program also offers a variety of networking and training activities, including summer schools and workshops. Each PhD student is co-supervised by one ELLIS **fellow/scholar** or **unit faculty** and one ELLIS **fellow/scholar**, **unit faculty** or **member** based in different European countries. Students conduct an exchange of at least 6 months with the international advisor during their degree. One of the advisors may also come from industry, in which case the student will collaborate closely with the industry partner, and spend min. 6 months conducting research

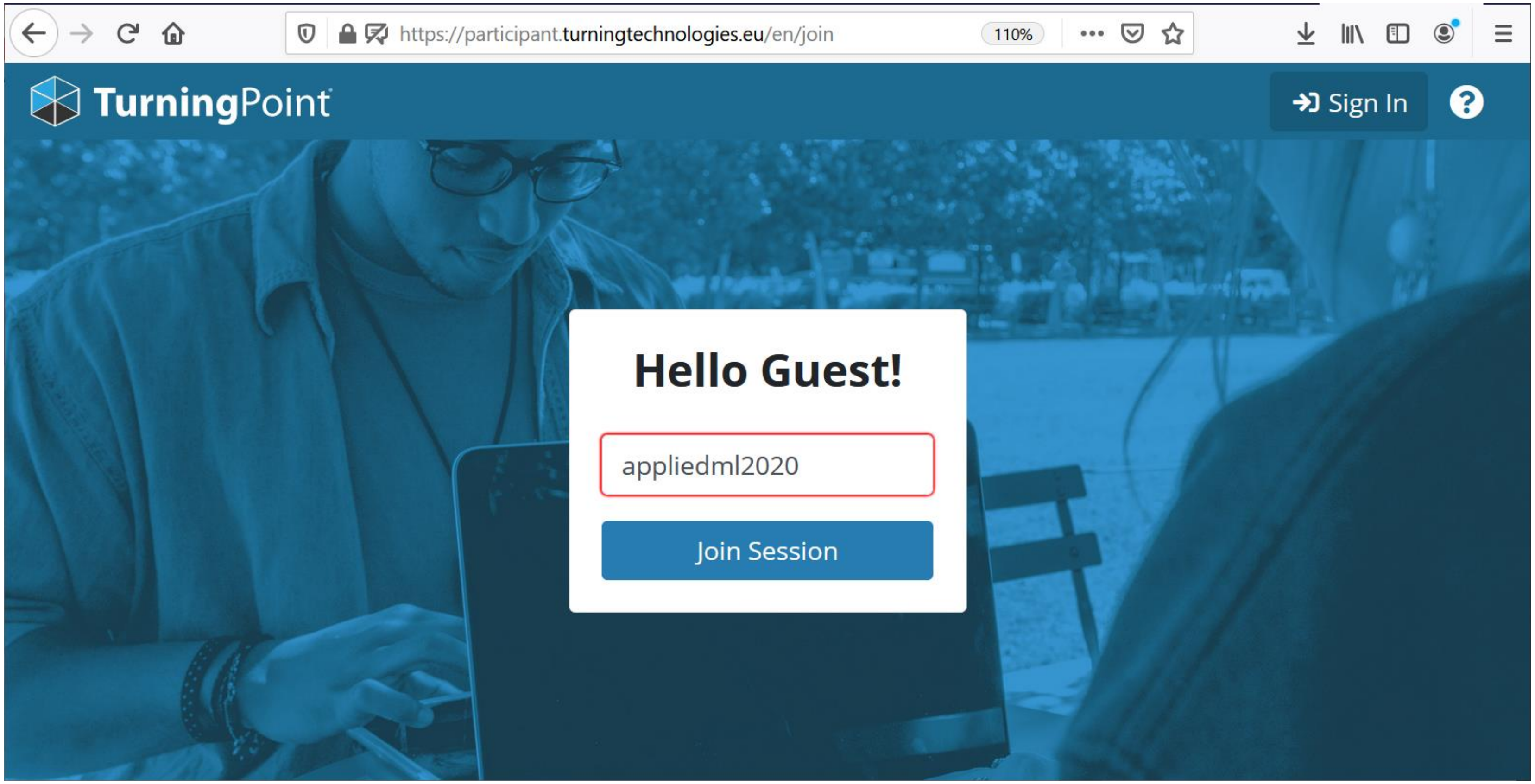
The call for applications to join as a PhD student in 2024 is open. Apply now!

**https://ellis.eu/news/ellis-phd-program-call-for-applications-2024**

# Launch polling system

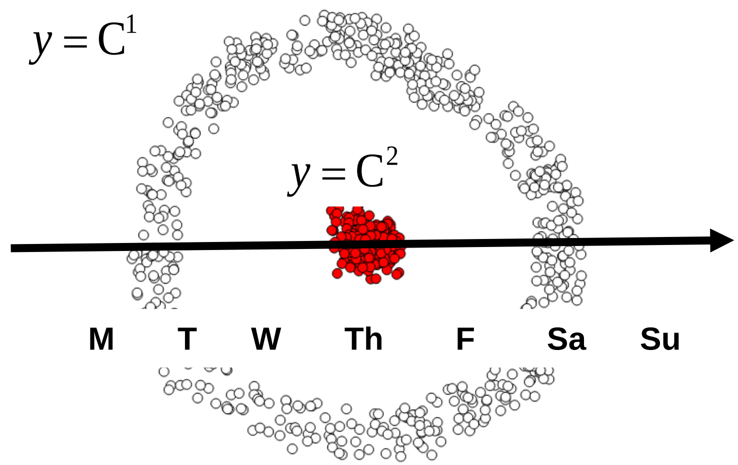
https://participant.turningtechnologies.eu/en/join

Acces as GUEST and enter the session id: *appliedml2020*



# Will our witch wear a spiky hat more often than flat hat ?

C1: flat hat




C2: spiky hat

**If we model the two classes with GMM + Bayes with one Gaussian to model each class**


**Decision Boundary**

$$p\left(y = C^1 \mid x\right) = p\left(y = C^2 \mid x\right)$$

$$p(x \mid y = C) \sim p\left(x \mid \mu^C, \Sigma^C\right), \ \Sigma^C = \begin{bmatrix} \sigma^C & 0 \\ 0 & \sigma^C \end{bmatrix}$$

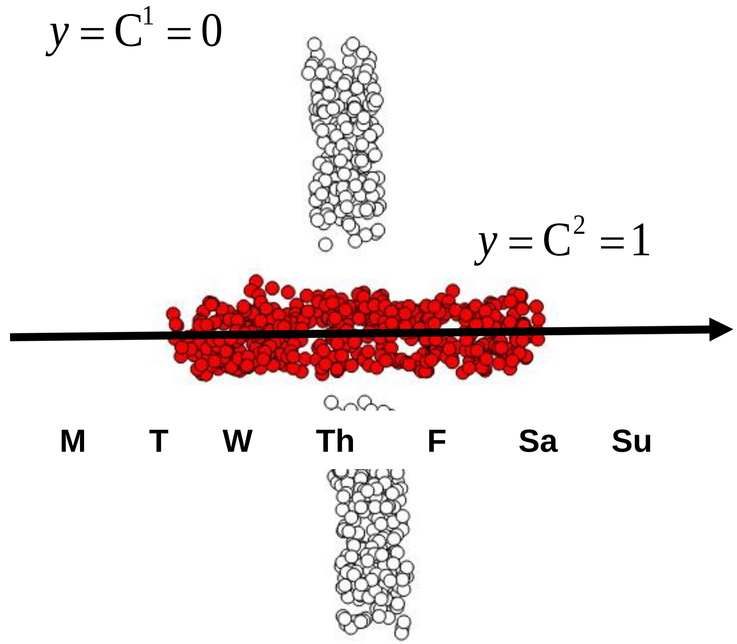
$\mu_c, \Sigma_c$ : mean and covariance matrix

A. C1

B. C2

# Which broom will our witch use most frequently?

C1: Broom 1




C2: Broom 2


If we use GMM + Bayes with one Diagonal Gaussian for each class

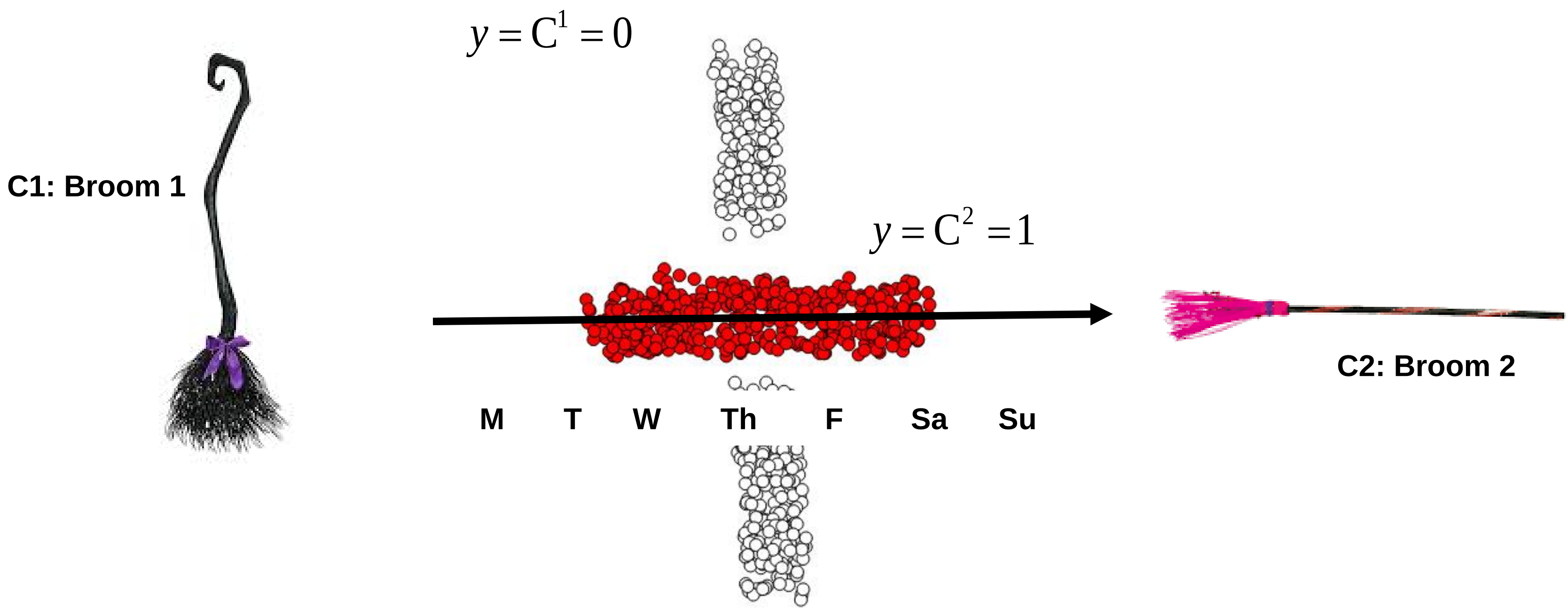
$$y = C) \sim p\left(x \mid \mu^c, \Sigma^c\right), \Sigma^c = \begin{bmatrix} \sigma_1^C & 0 \\ 0 & \sigma_2^C \end{bmatrix}$$

$\mu^c, \Sigma^c$ : mean and covariance matrix

Decision Boundary

$$p\left(y = C^1 \mid x\right) = p\left(y = C^2 \mid x\right)$$

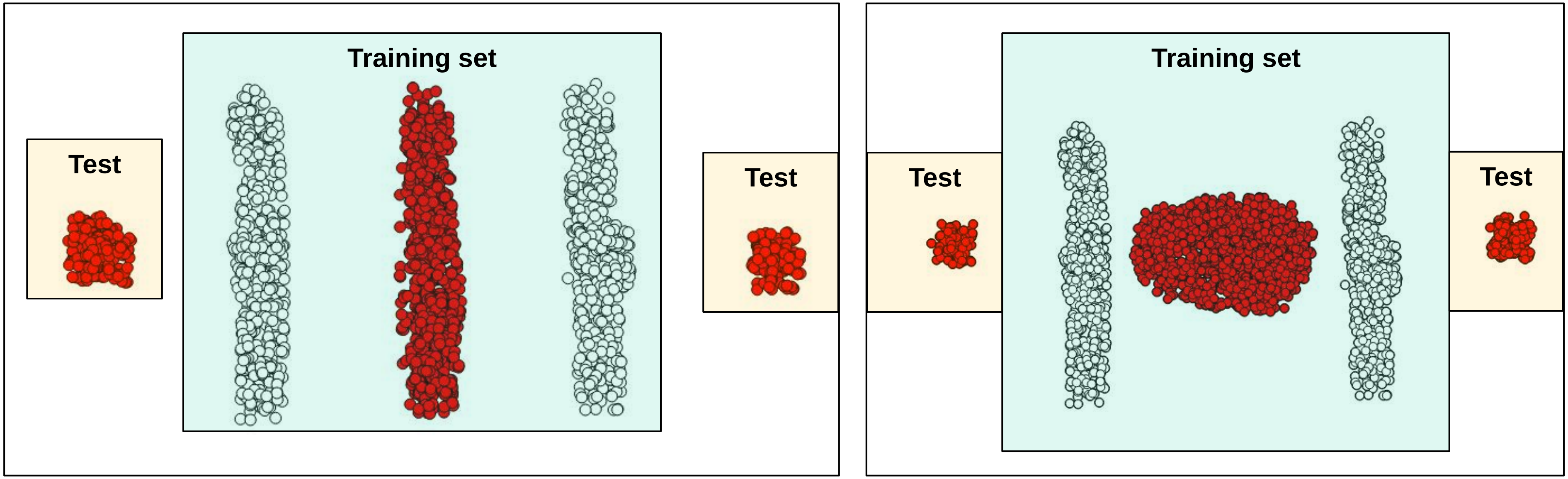
# Will the answer be the same if we were to use other covariance matrices?




A. Yes

B. No

Can we classify the outer datapoints using only the middle points for training?




(a) (b)

A. Yes for both a and b
B. No for both a and b
C. Yes for a only
D. Yes for b only
E. I do not know

# Determining the boundary across two pdf-s

We must determine the class with class label c that is most likely to have generated the datapoint $x$: $p(y=C\,|\,x)$

Bayes's rule: $p(y=C\,|\,x)=\dfrac{p(y=C)\,p(x\,|\,y=C)}{p(x)}$

$p(y=C)$: Probability of class C

$p(x)$: Marginal on $x$

$p(x\,|\,y=C)$: class conditional distribution of $x$
~ how the samples are distributed within class C.

$$p\left(x\,|\,y=C^1\right)\sim p\left(x\,|\,\mu_1,\Sigma_1\right)=\frac{1}{\left(2\pi\right)^{N/2}\left|\Sigma_1\right|^{1/2}}e^{-(x-\mu_1)(\Sigma_1)^{-1}(x-\mu_1)^T}$$

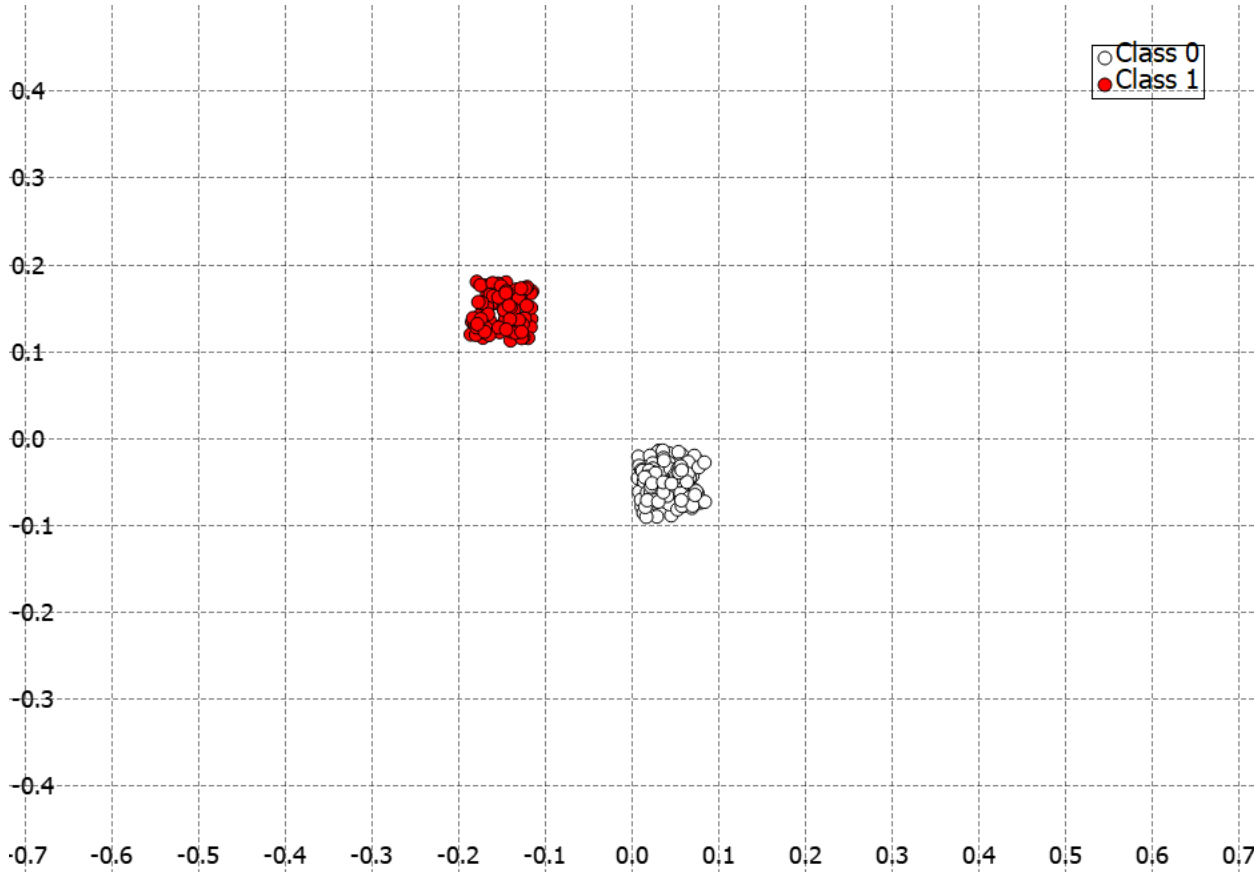
To determine the class label, compute optimal *Bayes classifier.*

A point $x$ belongs to class $C^1$ if $p\left(y=C^1\,|\,x\right)>p\left(y=C^2\,|\,x\right)$

Assuming equal class distribution, $p\left(y=C^1\right)=p\left(y=C^2\right)$ & $\ln p\left(y=C^1\right)=\ln p\left(y=C^2\right)$

$$\Leftrightarrow\ \left(x-\mu^1\right)^T\left(\Sigma^1\right)^{-1}\left(x-\mu^1\right)+\log\left|\Sigma^1\right|<\left(x-\mu^2\right)^T\left(\Sigma^2\right)^{-1}\left(x-\mu^2\right)+\log\left|\Sigma^2\right|$$

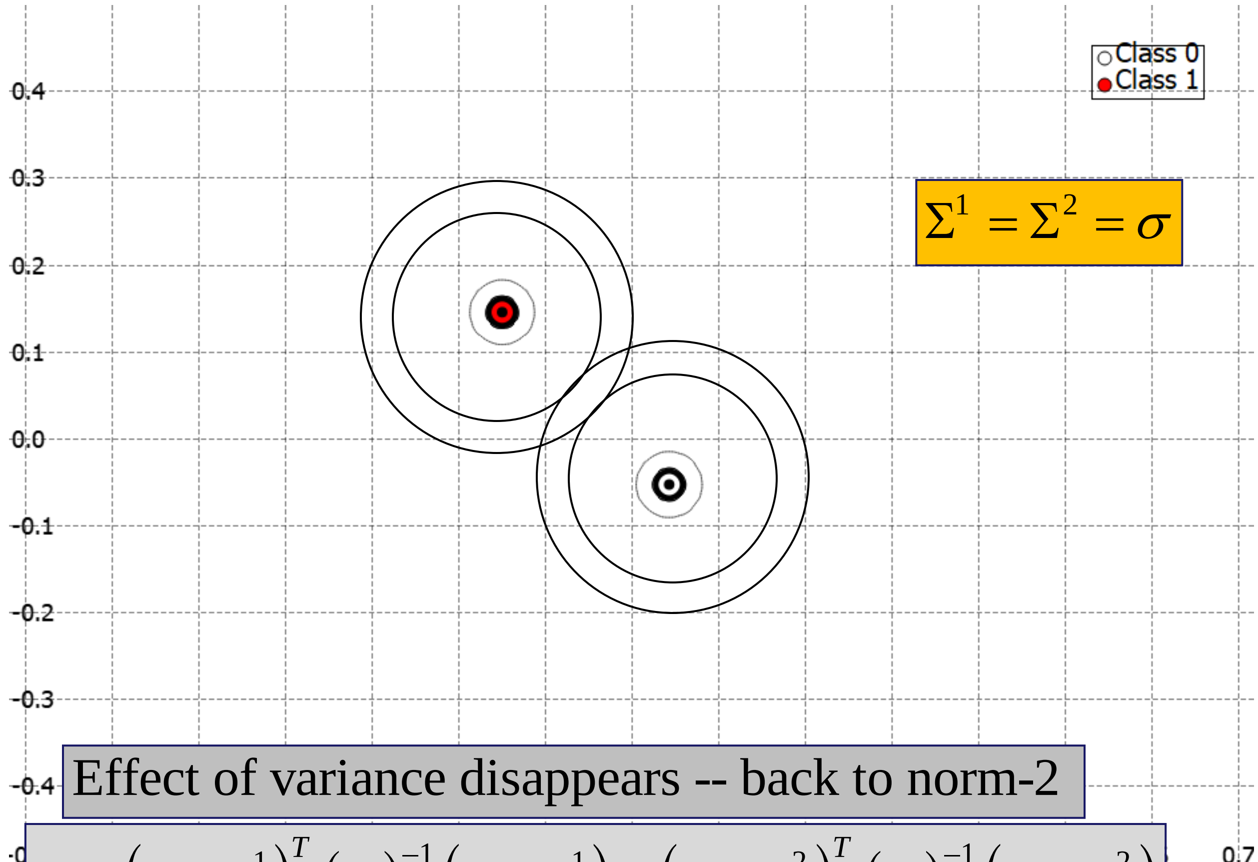
# Gaussian Discriminant Rule




$$\Leftrightarrow \left(x-\mu^1\right)^T\left(\Sigma^1\right)^{-1}\left(x-\mu^1\right)+\log\left|\Sigma^1\right|<\left(x-\mu^2\right)^T\left(\Sigma^2\right)^{-1}\left(x-\mu^2\right)+\log\left|\Sigma^2\right|$$

EPFL

# Gaussian Discriminant Rule



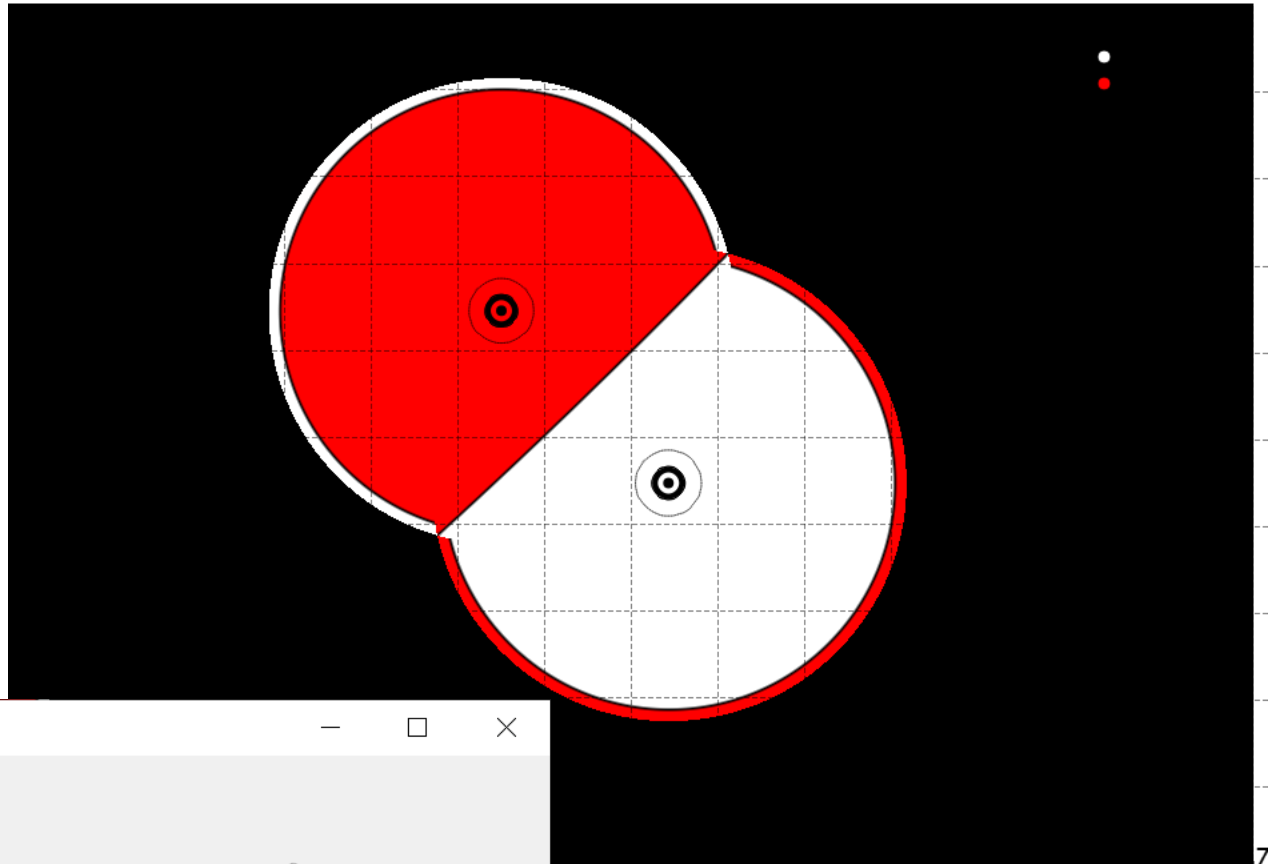

$$\Sigma^1 = \Sigma^2 = \sigma$$

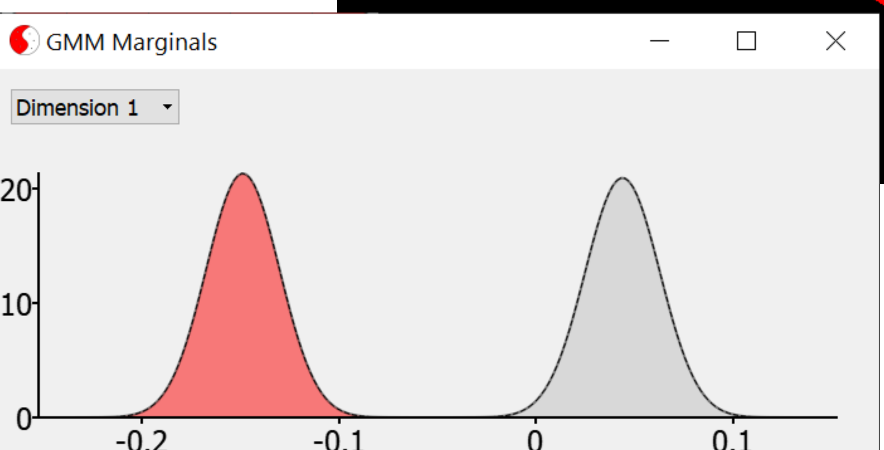
Effect of variance disappears -- back to norm-2

$$\Leftrightarrow \left(x-\mu^1\right)^T (\sigma)^{-1}\left(x-\mu^1\right) < \left(x-\mu^2\right)^T (\sigma)^{-1}\left(x-\mu^2\right)$$

$$\Leftrightarrow \left(x-\mu^1\right)^T \left(\Sigma^1\right)^{-1}\left(x-\mu^1\right) + \log\left|\Sigma^1\right| < \left(x-\mu^2\right)^T \left(\Sigma^2\right)^{-1}\left(x-\mu^2\right) + \log\left|\Sigma^2\right|$$

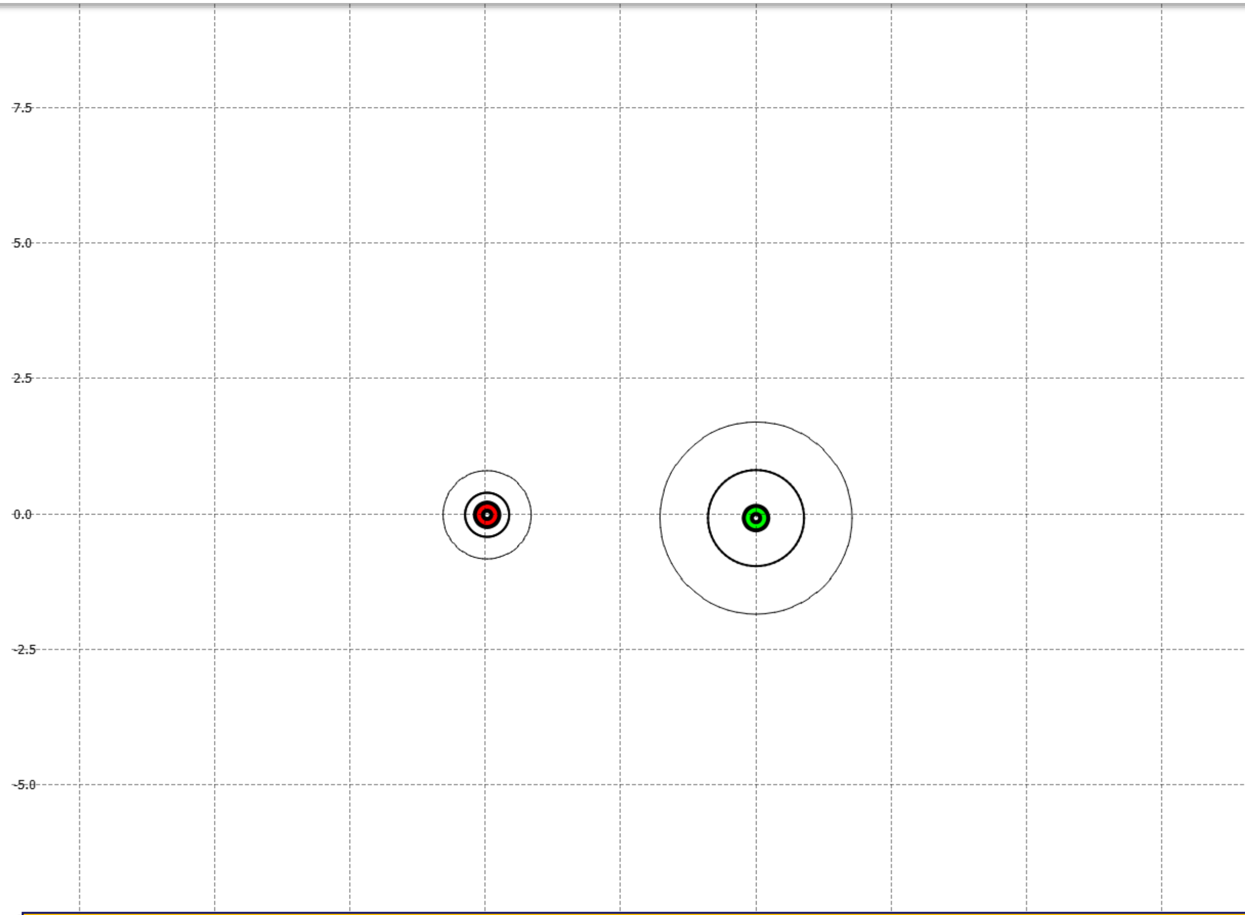
# Gaussian Discriminant Rule





# Contest

**Find the boundary when using GMM with one Gauss fct for each class**




**Winner gets a sweet**

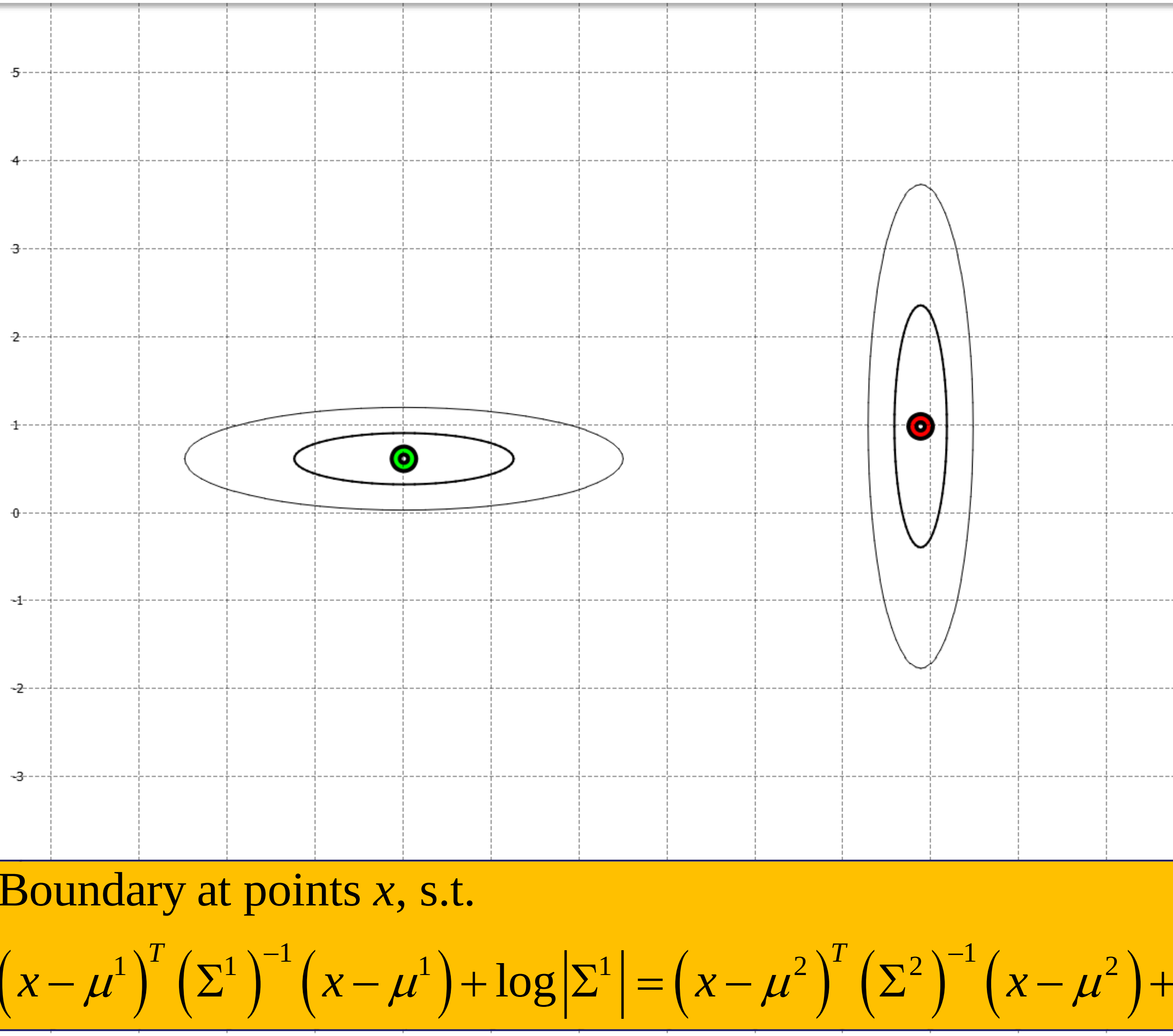
Boundary at points $x$, s.t.

$$\left(x-\mu^{1}\right)^{T}\left(\Sigma^{1}\right)^{-1}\left(x-\mu^{1}\right)+\log\left|\Sigma^{1}\right|=\left(x-\mu^{2}\right)^{T}\left(\Sigma^{2}\right)^{-1}\left(x-\mu^{2}\right)+\log\left|\Sigma^{2}\right|$$

EPFL

# Contest

Draw the boundary when using GMM with one Gauss fct for each class




Winner gets a sweet

Boundary at points $x$, s.t.

$$\left(x-\mu^1\right)^T\left(\Sigma^1\right)^{-1}\left(x-\mu^1\right)+\log\left|\Sigma^1\right|=\left(x-\mu^2\right)^T\left(\Sigma^2\right)^{-1}\left(x-\mu^2\right)+\log\left|\Sigma^2\right|$$

# Nonlinearity of the Decision Boundary

Recall: to determine the class label, compute optimal *Bayes classifier.*

$$\boxed{\text{A point } x \text{ belongs to class } C^1 \text{ if } p\left(y=C^1 \mid x\right) > p\left(y=C^2 \mid x\right)}$$

$$\Rightarrow \ln p\left(y=C^1 \mid x\right) > \ln p\left(y=C^2 \mid x\right)$$

Consider the univariate case (1D data) and classes equally likely $p\left(y=C^1\right)=p\left(y=C^2\right)$

$$\frac{(x-\mu_1)^2}{2\sigma_1^2}+\ln\sqrt{2\pi}\sigma_1=\frac{(x-\mu_1)^2}{2\sigma_2^2}+\ln\sqrt{2\pi}\sigma_2$$

$$\frac{x^2-2x\mu_1+\mu_1^2}{2\sigma_1^2}-\frac{x^2-2x\mu_2+\mu_2^2}{2\sigma_2^2}+\ln\sqrt{2\pi}(\sigma_1-\sigma_2)=0$$

$$\boxed{\left(\frac{\sigma_2^2}{2\sigma_1^2\sigma_2^2}-\frac{\sigma_1^2}{2\sigma_1^2\sigma_2^2}\right)x^2-\left(\frac{\sigma_2^2\mu_1-\sigma_1^2\mu_2}{\sigma_1^2\sigma_2^2}\right)x+\frac{\sigma_2^2\mu_1^2-\sigma_1^2\mu_2^2}{2\sigma_1^2\sigma_2^2}+\ln\sqrt{2\pi}(\sigma_1-\sigma_2)=0}$$

Quadratic equation → Nonlinear boundary

# Nonlinearity of the Decision Boundary

The decision boundary has the form:

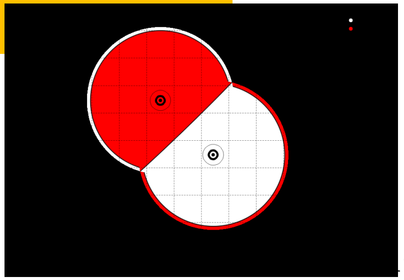
$$\left.\begin{array}{l} ax^2 + bx + c = 0 \rightarrow \text{in the univariate case} \\ \mathrm{x}^T A x + b^T x + c = 0 \rightarrow \text{in the multivariate case} \end{array}\right\} \text{Quadratic Discriminant Analysis}$$

In the case where $\sigma_1^2 = \sigma_2^2$ or $\Sigma_1 = \Sigma_2$ for the multivariate case

$$\cancelto{0}{\left(\frac{\sigma_2^2}{2\sigma_1^2\sigma_2^2} - \frac{\sigma_1^2}{2\sigma_1^2\sigma_2^2}\right)} x^2 - \left(\frac{\sigma_2^2\mu_1 - \sigma_1^2\mu_2}{\sigma_1^2\sigma_2^2}\right) x + \frac{\sigma_2^2\mu_1^2 - \sigma_1^2\mu_2^2}{2\sigma_1^2\sigma_2^2} + \ln\sqrt{2\pi}(\sigma_1 - \sigma_2) = 0$$

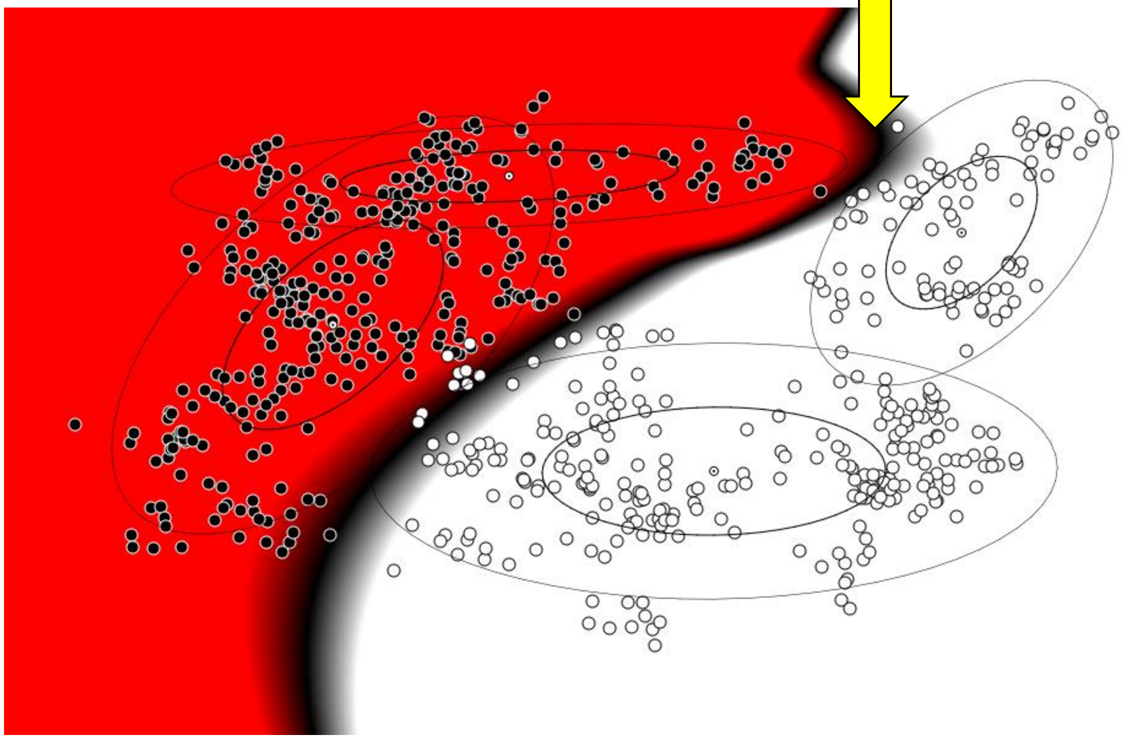
$$\left.\begin{array}{l} bx + c = 0 \rightarrow \text{in the univariate case} \\ b^T x + c = 0 \rightarrow \text{in the multivariate case} \end{array}\right\} \text{Linear Discriminat Analysis}$$

Linear equation → Linear boundary



# Classification with multiple Gauss fcts (GMM-s)

Where is the boundary?



Example of binary classification using
2 Gaussian Mixture Models with 2 Gauss functions each

# Maximum Likelihood Discriminant Rule for Multi-Class Classification

The maximum likelihood (ML) discriminant rule predicts the class of an observation $x$ using:

$$c(x) = \arg\max_{k=1\ldots K} p_k(x) \qquad \text{for} \quad K \text{ classes}$$

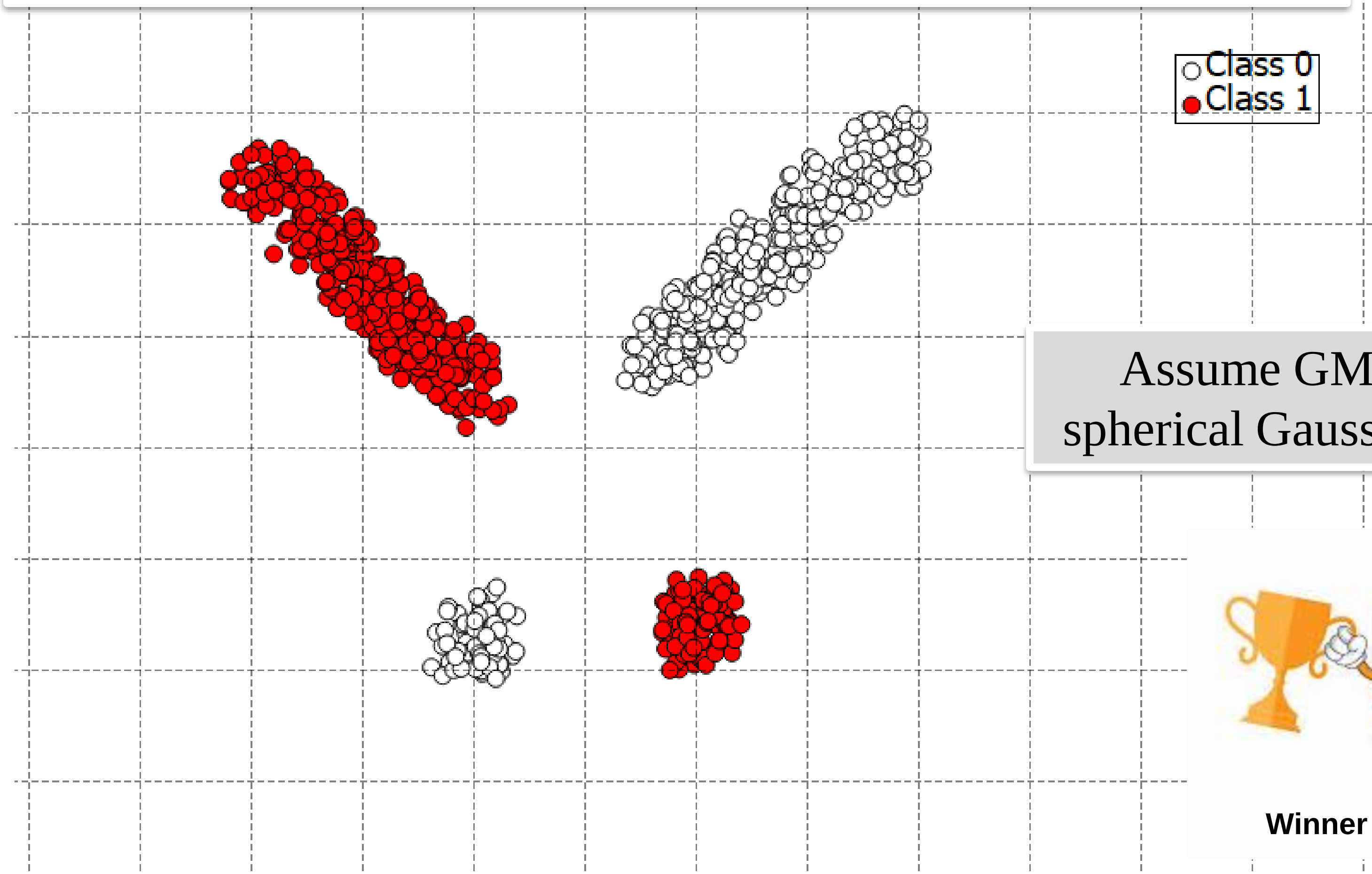
ML discriminant rule is minimum of minus the log-likelihood (equiv. to maximizing the likelihood):

$$C^k(x) = \arg\min_{k} \left\{ \left(x - \mu^k\right)\left(\Sigma^k\right)^{-1}\left(x - \mu^k\right)^T + \log\left|\Sigma^k\right| \right\}$$

Even though there are two classes, the groups being distinct, the boundaries can be thought of as a 4-class classification problem.

# Contest

Find the boundaries for the two distributions below




Assume GMM with spherical Gauss functions


$$C^k(x) = \arg\min_k \left\{ \left(x - \mu^k\right)\left(\Sigma^k\right)^{-1}\left(x - \mu^k\right)^T + \log\left|\Sigma^k\right| \right\}$$

# Gaussian ML Discriminant Rule and LDA

When all class densities have the same covariance matrix, $\Sigma^k = \Sigma$, the discriminant rule is linear. This can be turned into a single optimization problem for supervised learning when the class labels are known. This is known as Linear discriminant analysis (LDA).

$$0 < \left(x-\mu^1\right)(\Sigma)^{-1}\left(x-\mu^1\right)^T < \left(x-\mu^2\right)(\Sigma)^{-1}\left(x-\mu^2\right)^T$$

or equivalently

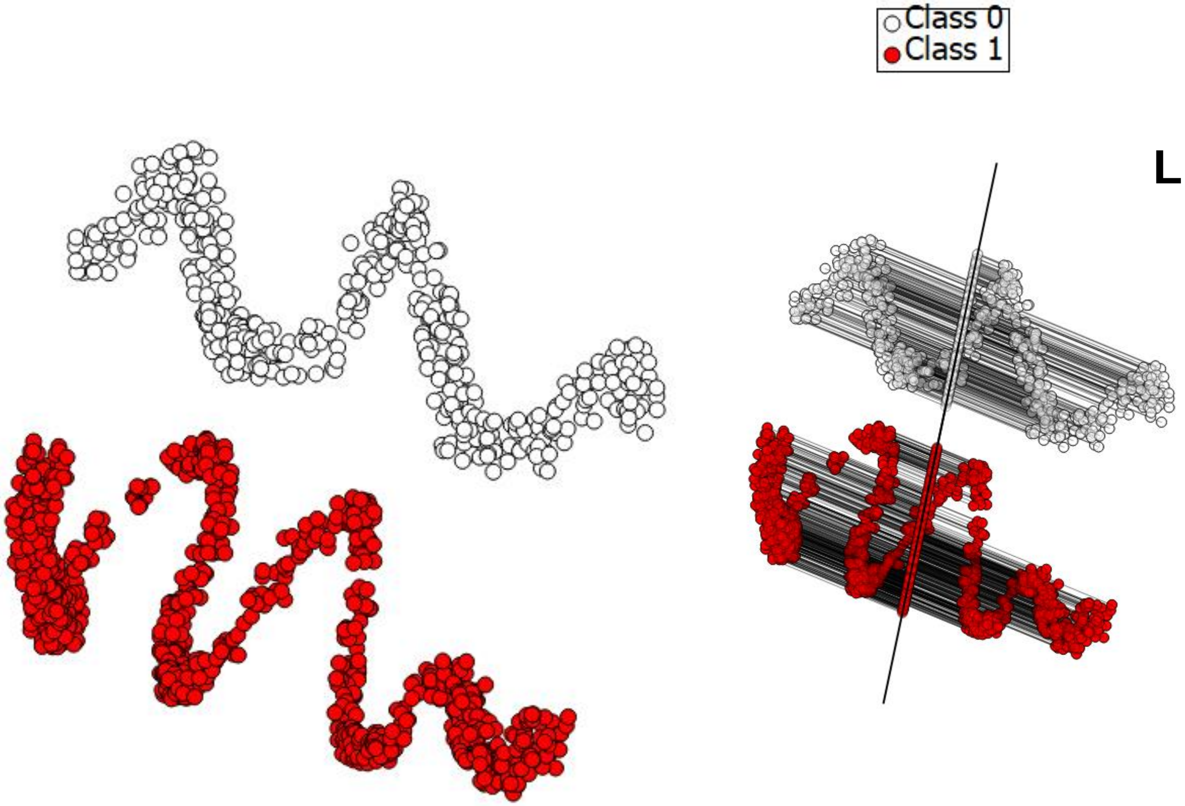
$$c_k(x) = \arg\min_{k=\{1,2\}} \left\{ \left(x-\mu^k\right)(\Sigma)^{-1}\left(x-\mu^k\right)^T \right\}$$

LDA can be used as projection technique, like PCA, using as objective function the discriminant rule.

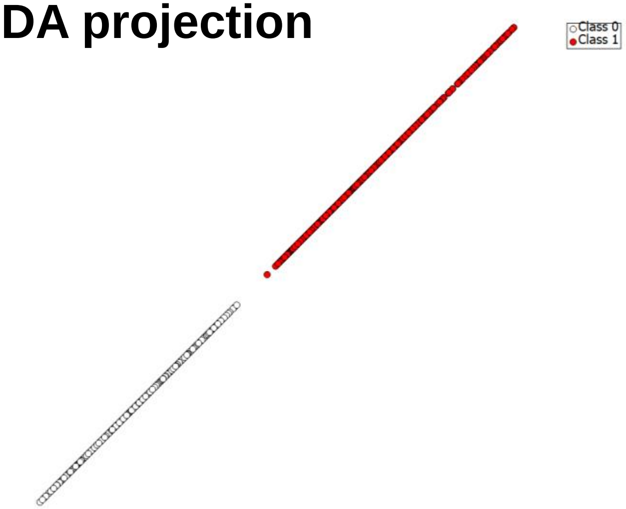
**It finds the projection that separates best the two classes.**

***See Lecture Notes for description of LDA.***

# Gaussian ML Discriminant Rule and LDA




**LDA projection**

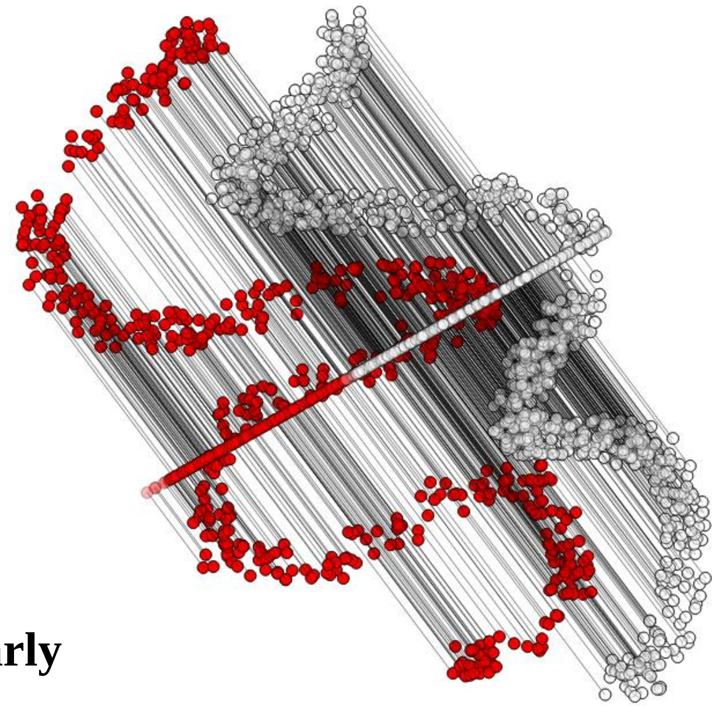



**1st PCA projection**

**2nd PCA projection**

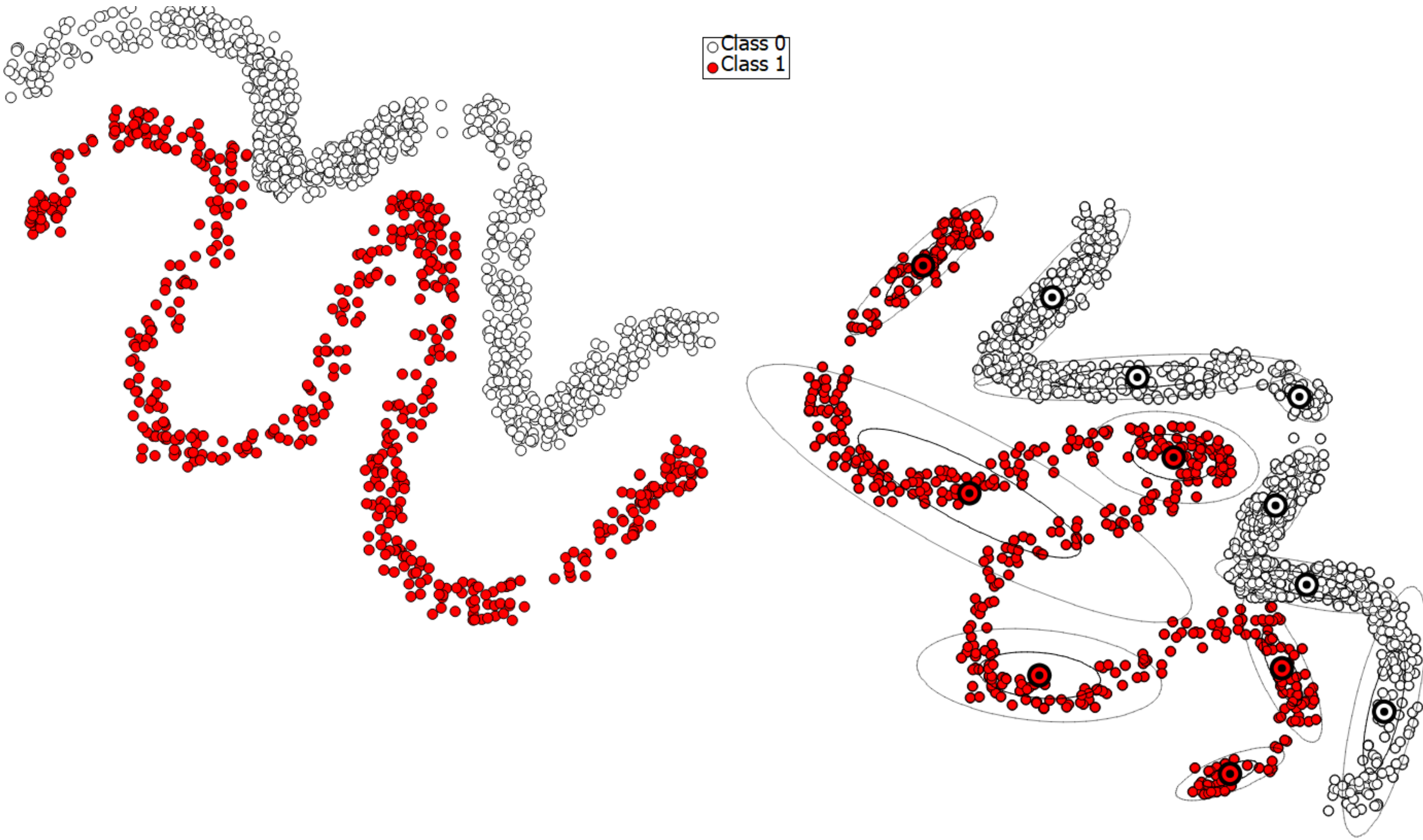
# Gaussian ML Discriminant Rule and LDA


LDA projection

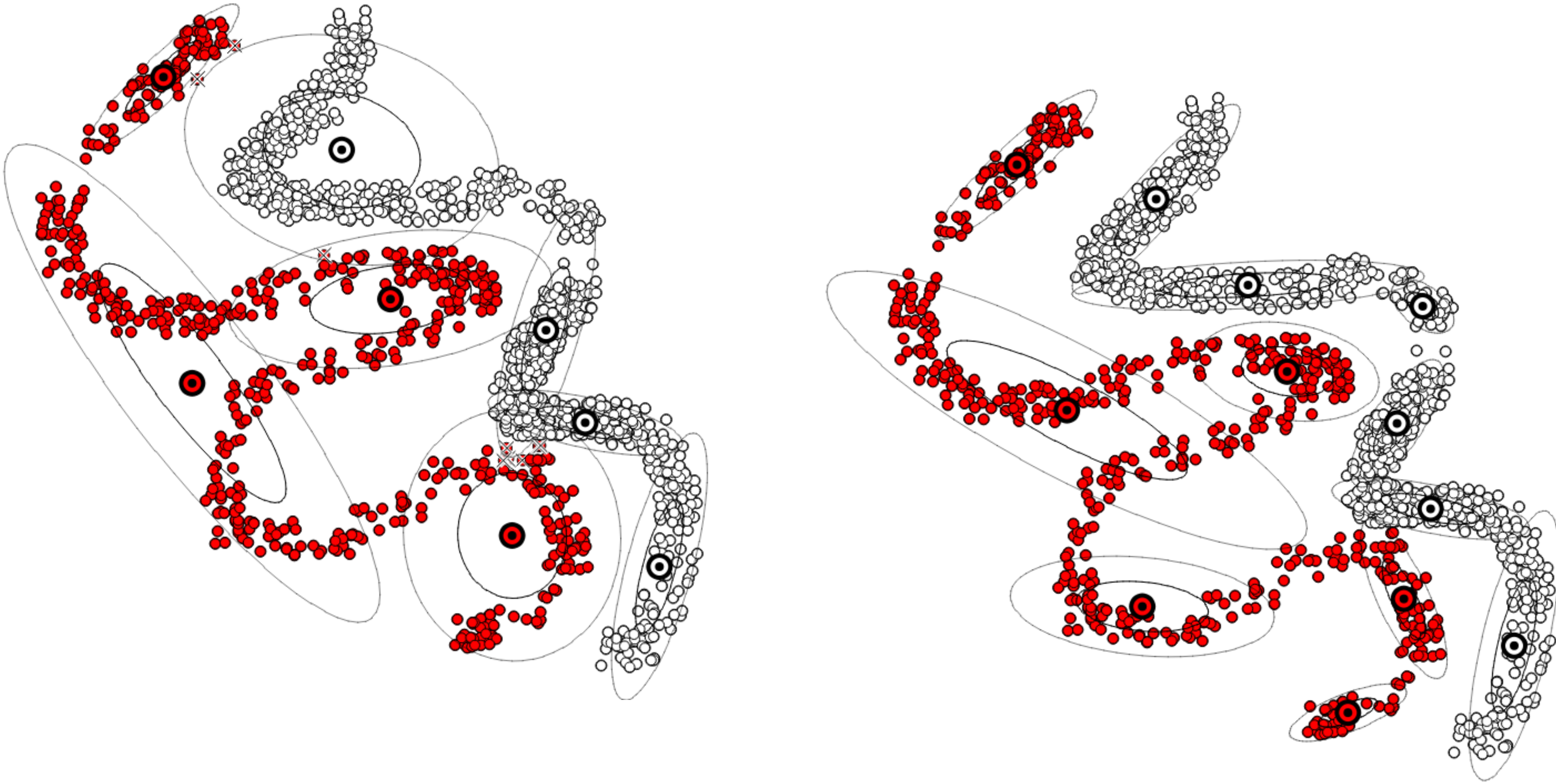


**LDA fails when the classes cannot be separated linearly**

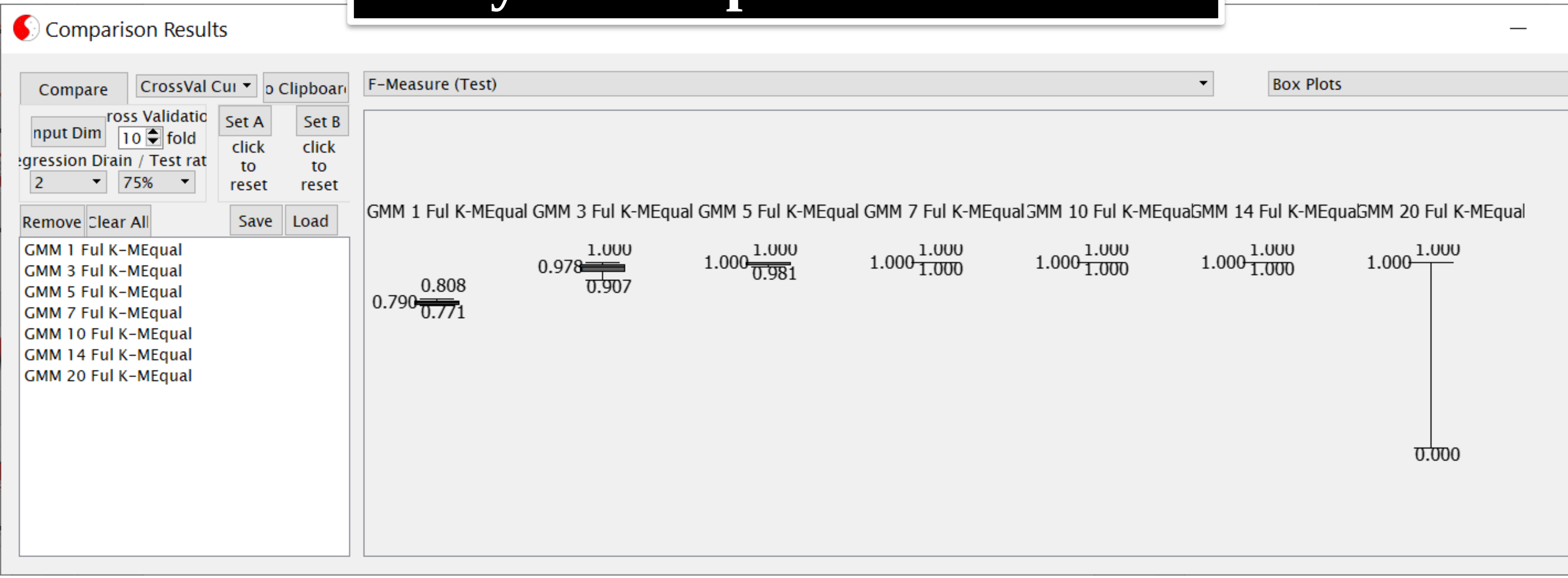
# GMM with multiple Gauss functions for each class



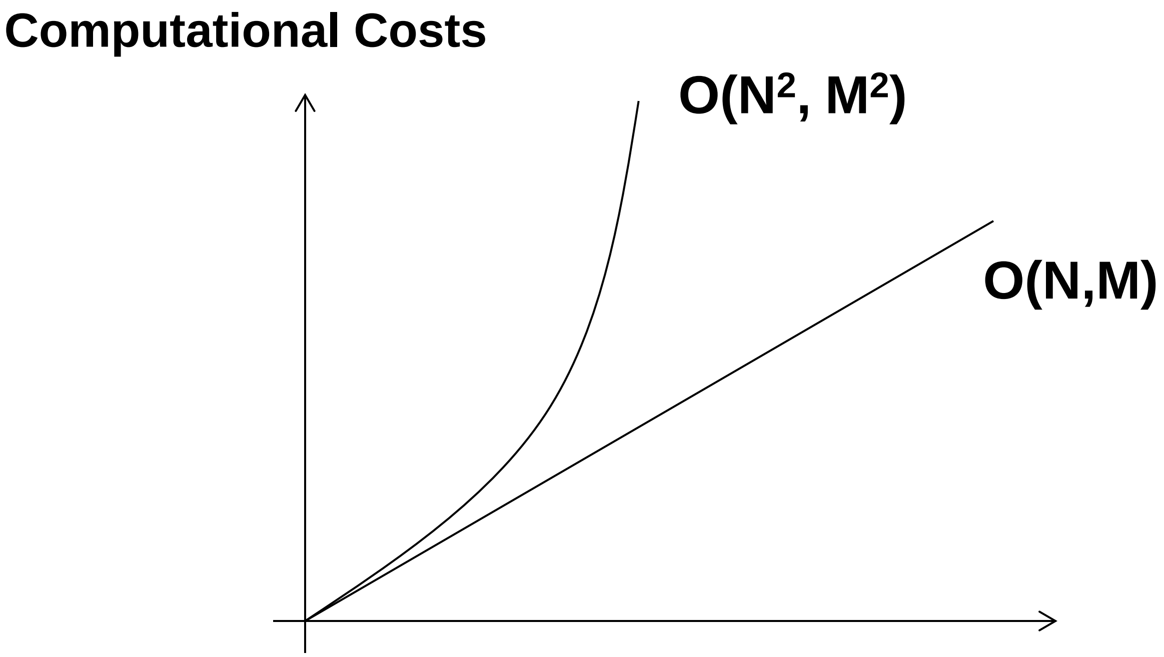
# How would you evaluate which model is the best?



# Can you interpret these results?



# Curse of Dimensionality

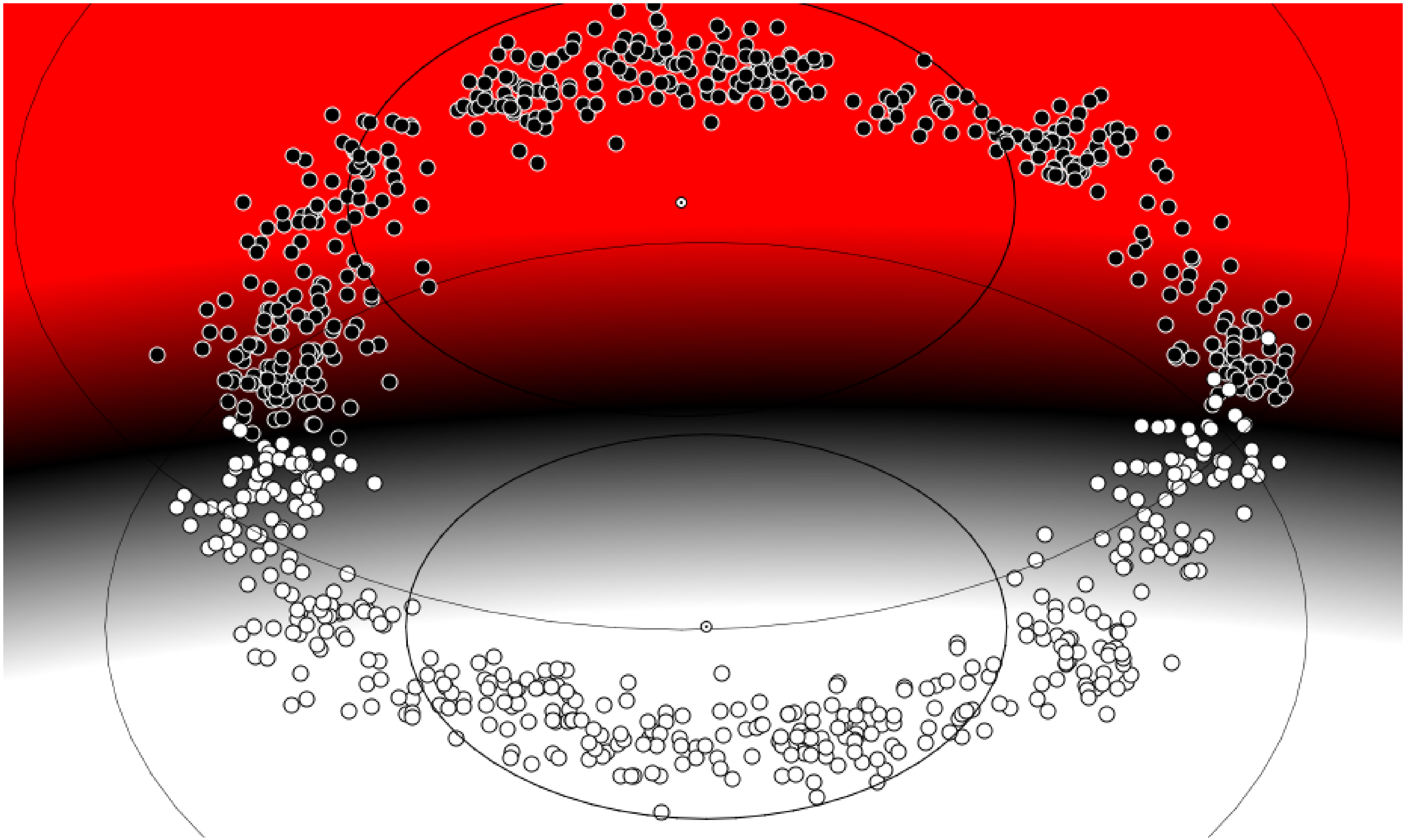



**N: Nb of dimensions**
**M: Nb of datapoints**

**Computational costs may grow as a function of number of dimensions or of number of datapoints**

# Example : Classification with 2 GMMs

**(1 Gaussian per model, spherical covariance matrix)**

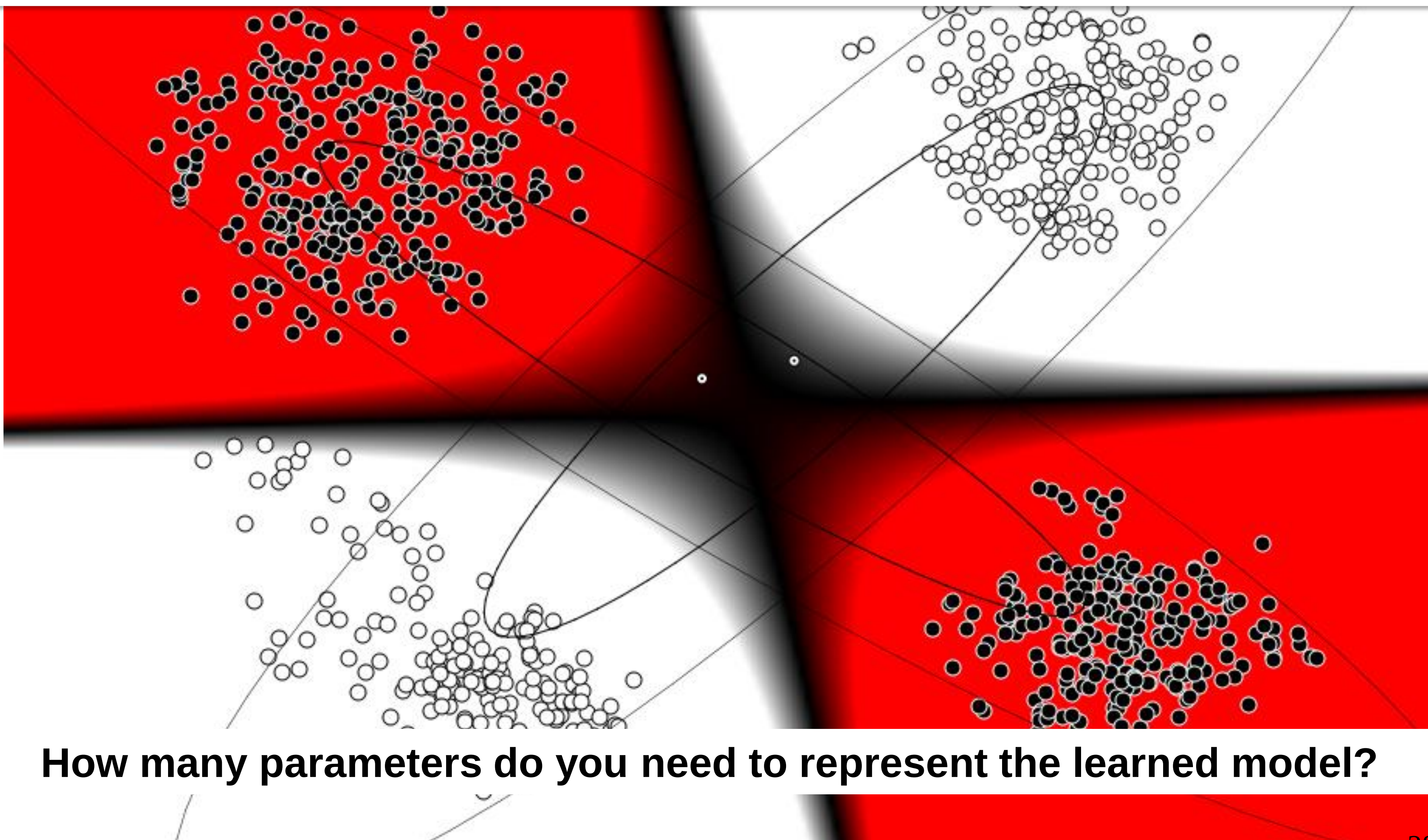


**How many parameters do you need to represent the learned model?**

# Example : Classification with 2 GMMs

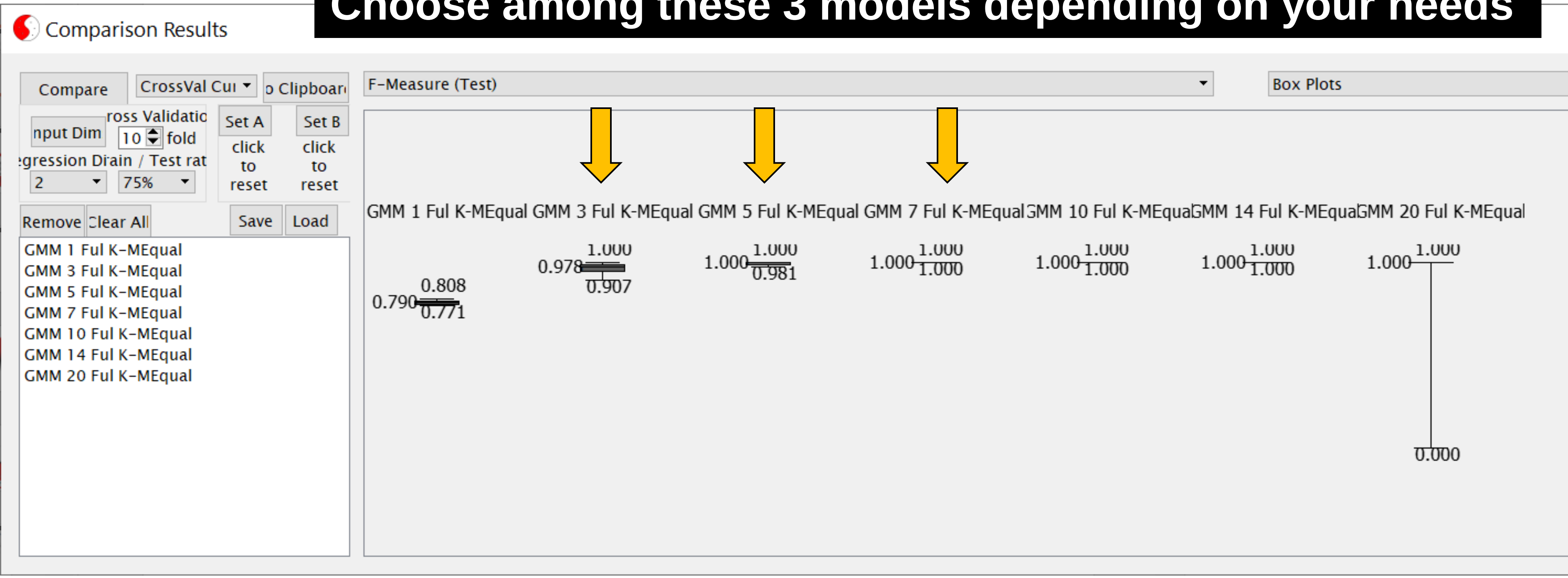
**(1 Gaussian per model, full covariance matrix)**

**Computational costs in GMM grow quadratically with N and linearly with M at training and quadratically with N at testing**

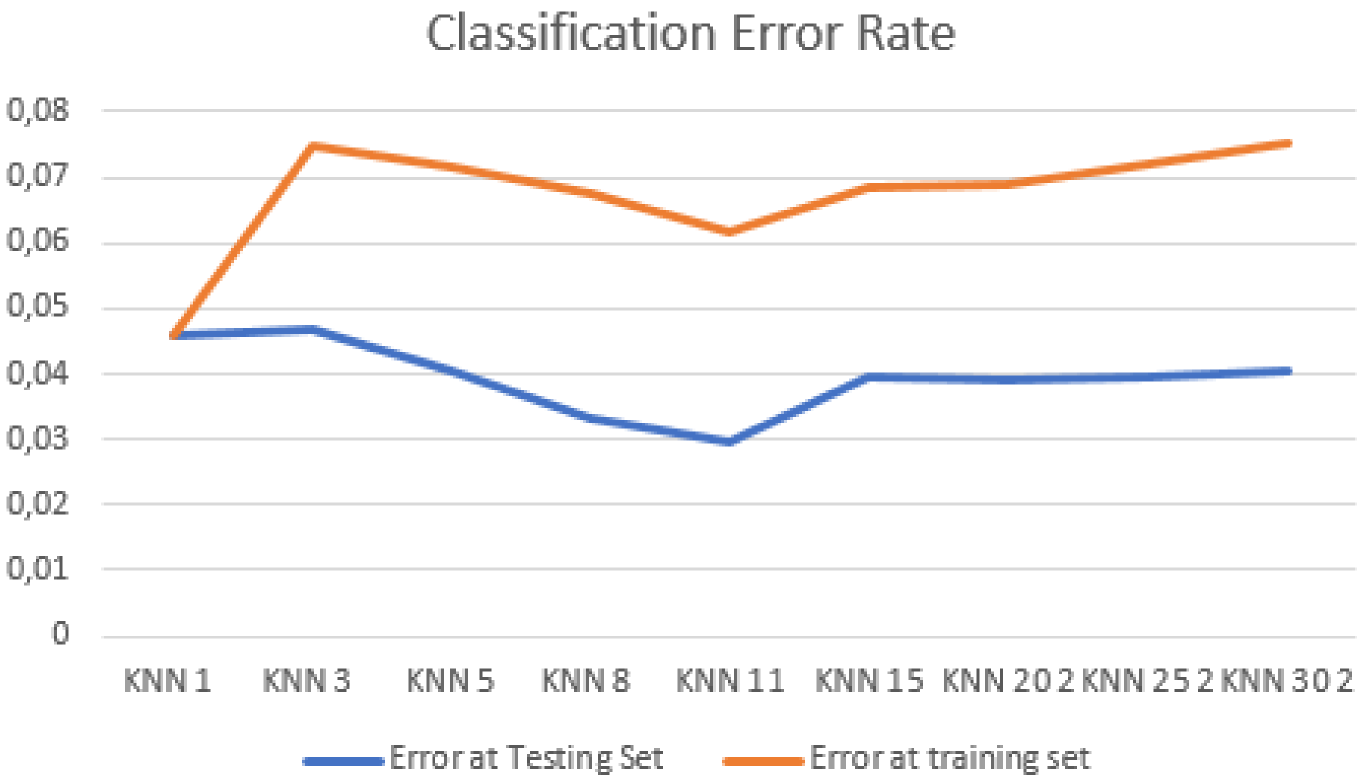


**How many parameters do you need to represent the learned model?**

# Choose among these 3 models depending on your needs



# Classification Error on training and testing sets




Which of these three combination is overfitting?

A. a
B. b
C. c

| | Training Error | Testing Error |
|---|---|---|
| a | Low | High |
| b | Low | Low |
| c | High | High |

# Crossvalidation & Choice of training/testing ratio

- Avoid **overfitting** (i.e. fitting too well all datapoints including noise)
  → Train the classifier with a small sample of all datapoints
  → Test the classifier with the remaining datapoints.

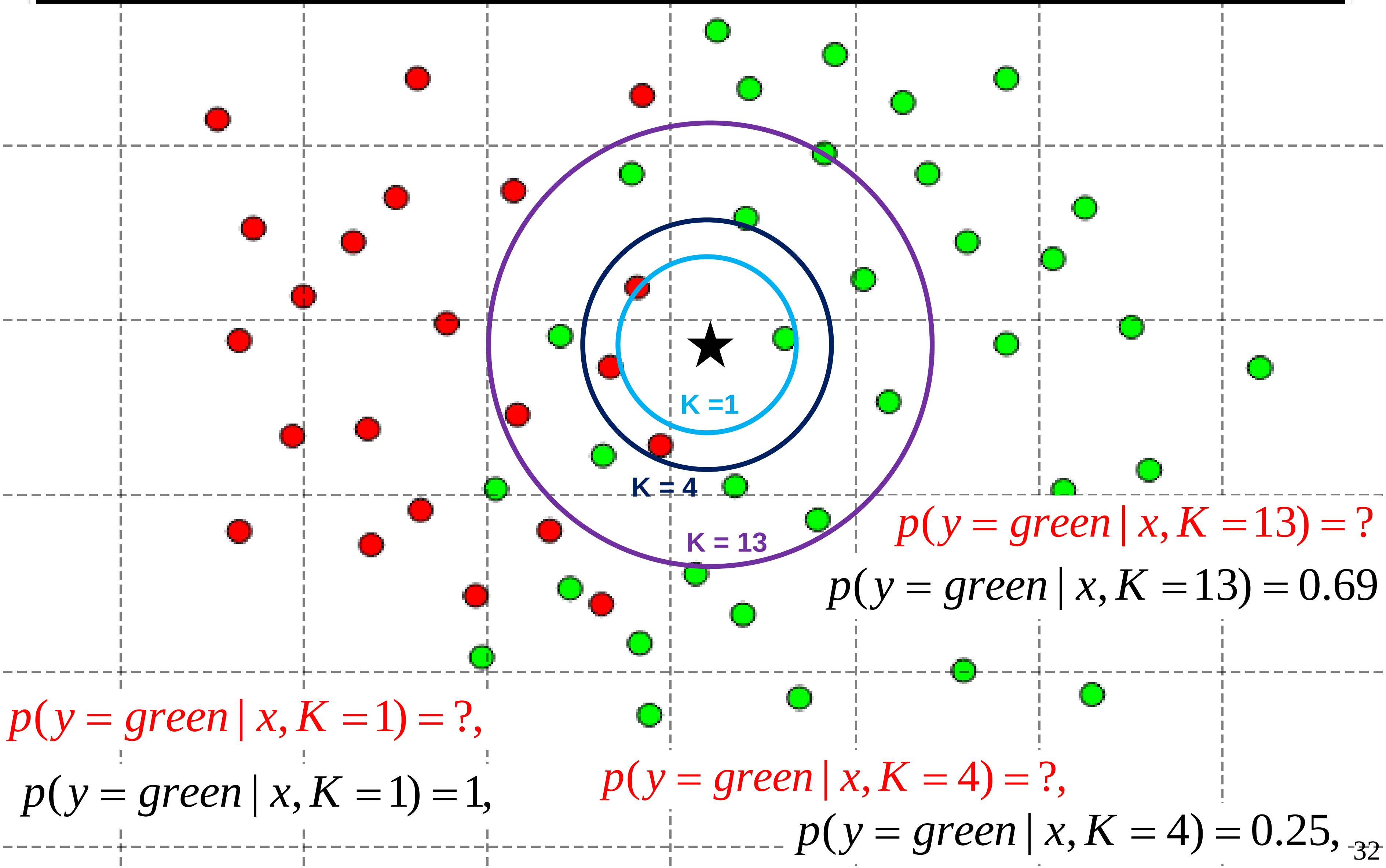
- Typical choice of training/testing set ratio is $2/3^{rd}$ training, $1/3^{rd}$ testing.

- The smaller the ratio, the more robust the classification

- **Several-fold crossvalidation**

- Typical choice is 10-fold crossvalidation. However, this depends on how many datapoints you have in your dataset!

# Classification with K- Nearest Neighbors (K-NN)

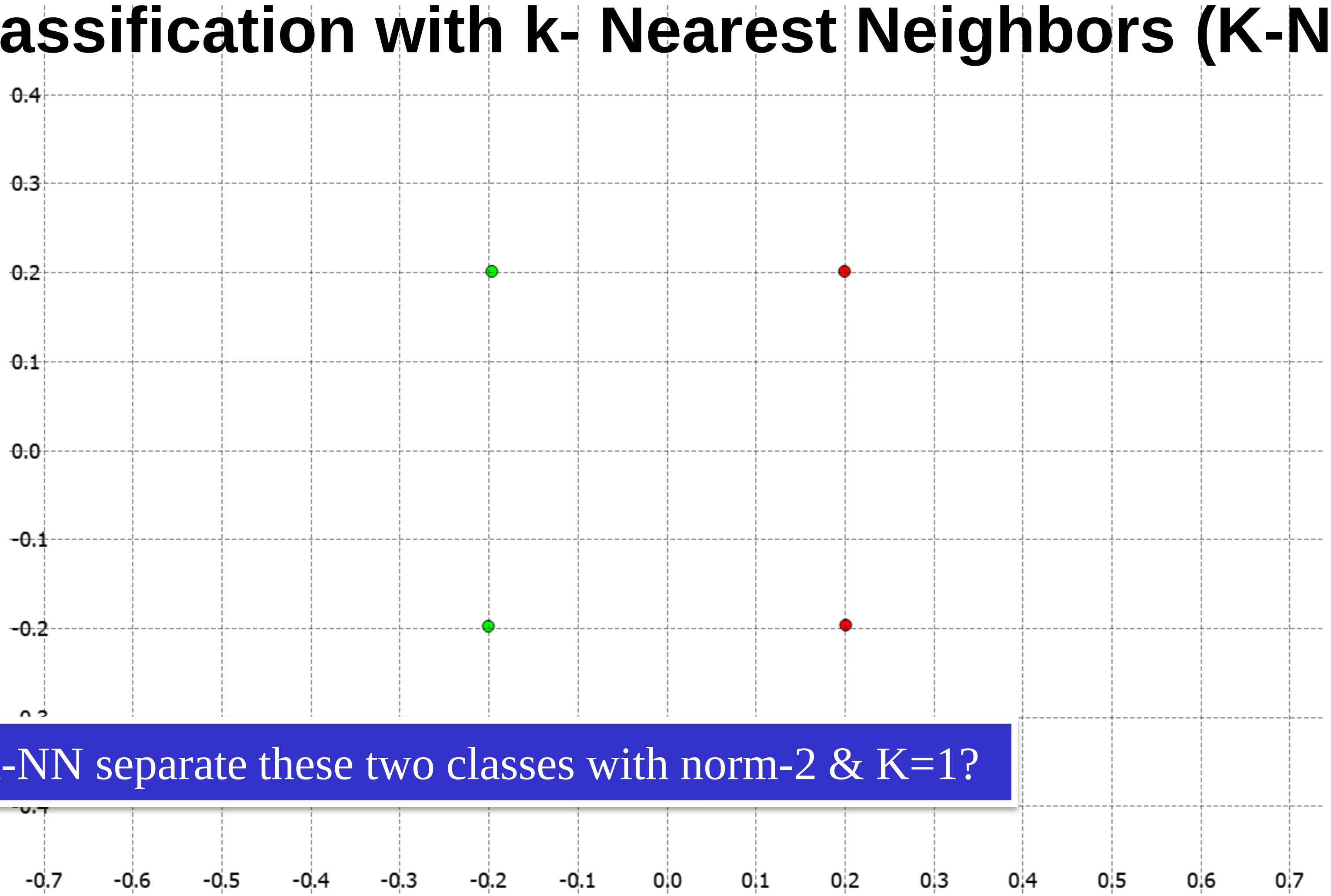



$$p(y = green \mid x, K = 13) = ?$$
$$p(y = green \mid x, K = 13) = 0.69$$

$$p(y = green \mid x, K = 1) = ?,$$
$$p(y = green \mid x, K = 1) = 1,$$

$$p(y = green \mid x, K = 4) = ?,$$
$$p(y = green \mid x, K = 4) = 0.25,$$

# Classification with k- Nearest Neighbors (K-NN)



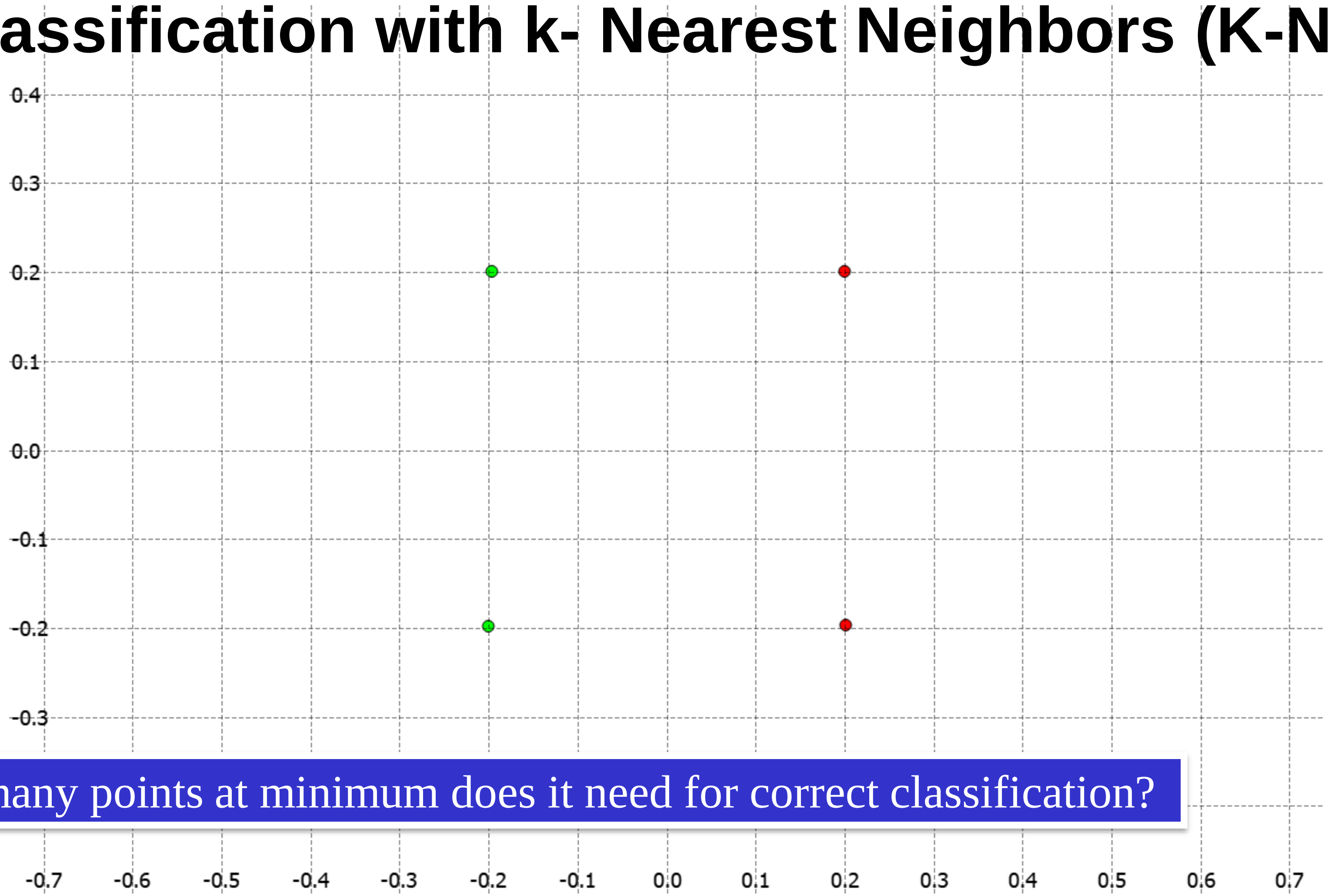

Can K-NN separate these two classes with norm-2 & K=1?

A. YES

B. NO

C. I do not know

# Classification with k- Nearest Neighbors (K-NN)



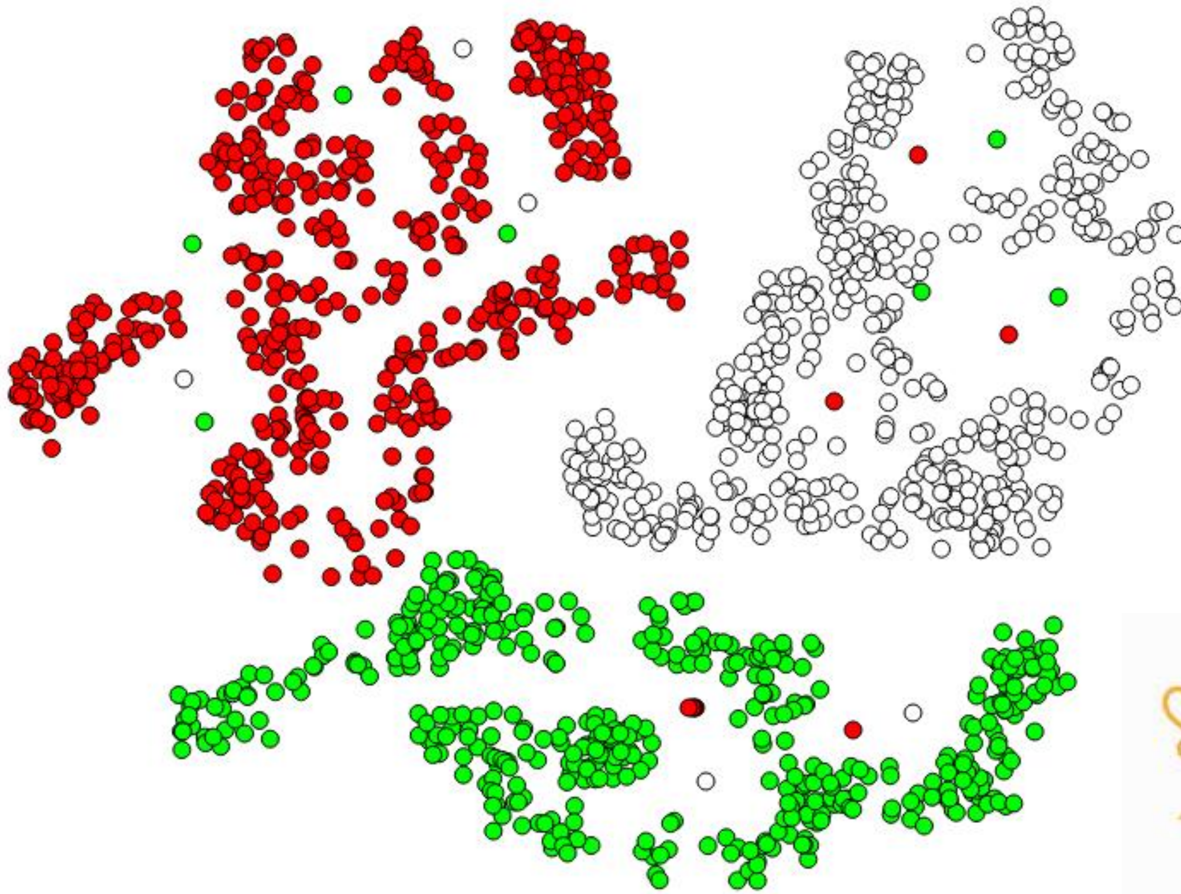

How many points at minimum does it need for correct classification?

A. 2

B. 3

C. 4

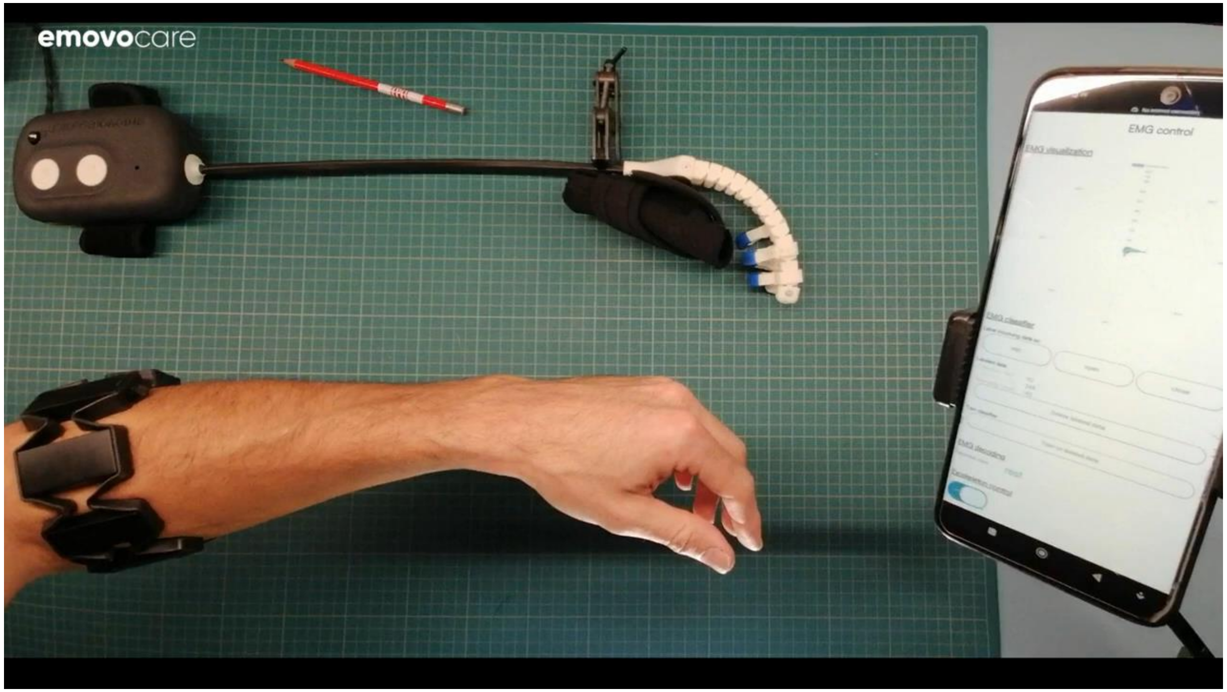
# Draw the boundaries found by KNN for different K




Winner gets a sweet

**K=1, K=3, K=20**

# KNN application – Training EMG controller




**EMG control App : provides visualization and a way to collect labelled data rapidly**

**Training phase (~30 sec) : User input from 8D EMG -> labelled into 3 classes (closed, open, rest) using the app**

**Control phase : Real-time KNN classification to control exoskeleton accordingly**