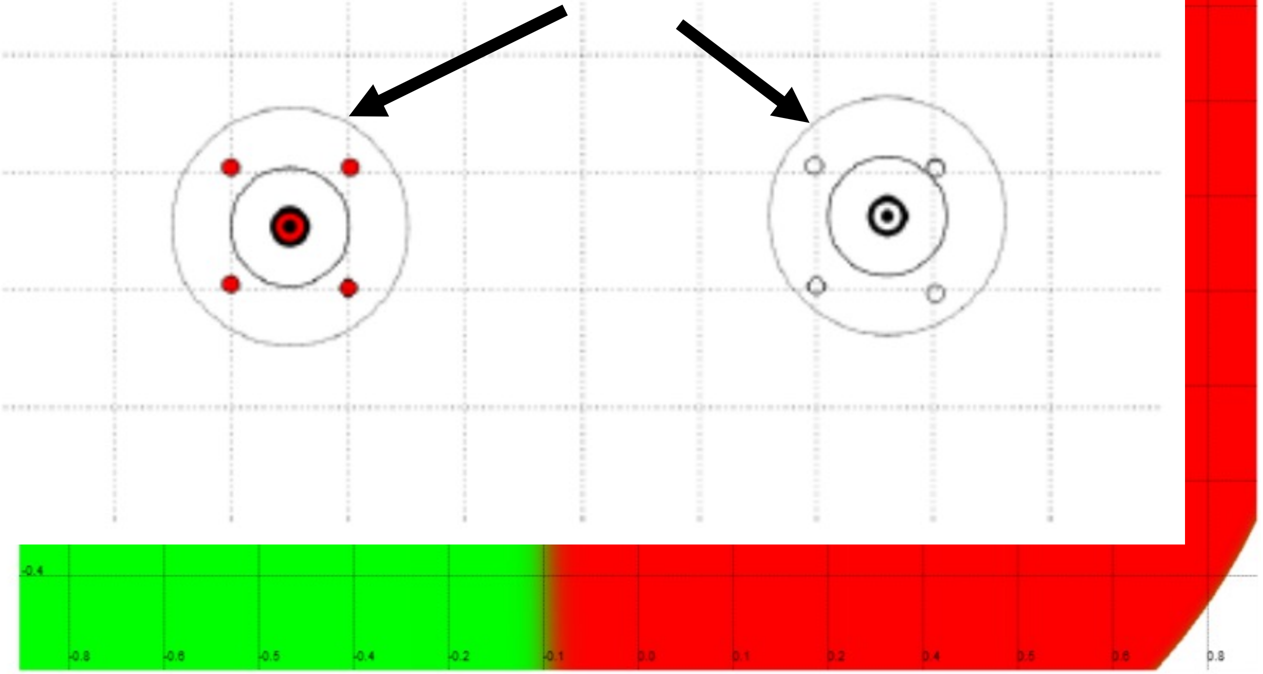
Mixture of Gaussians and Bayes' classification

❖A) a)

- Covariance matrix identical
- Centers at the same height
- Same number of points

Gaussian not normalized: the isolines represent the same probability !



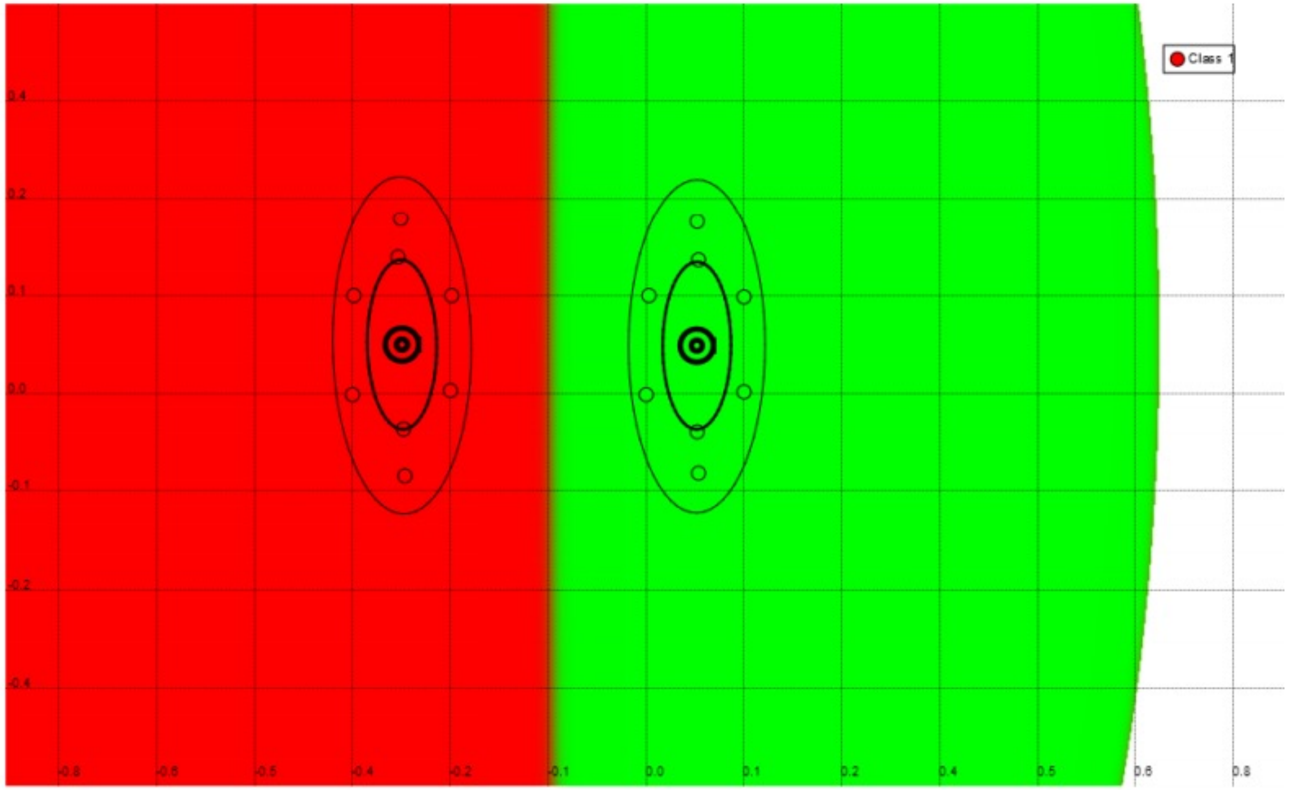
Linear Boundary equidistant from the two centers

Mixture of Gaussians and Bayes' classification

❖A) b)

- Covariance matrix identical
- Centers at the same height
- Same number of points

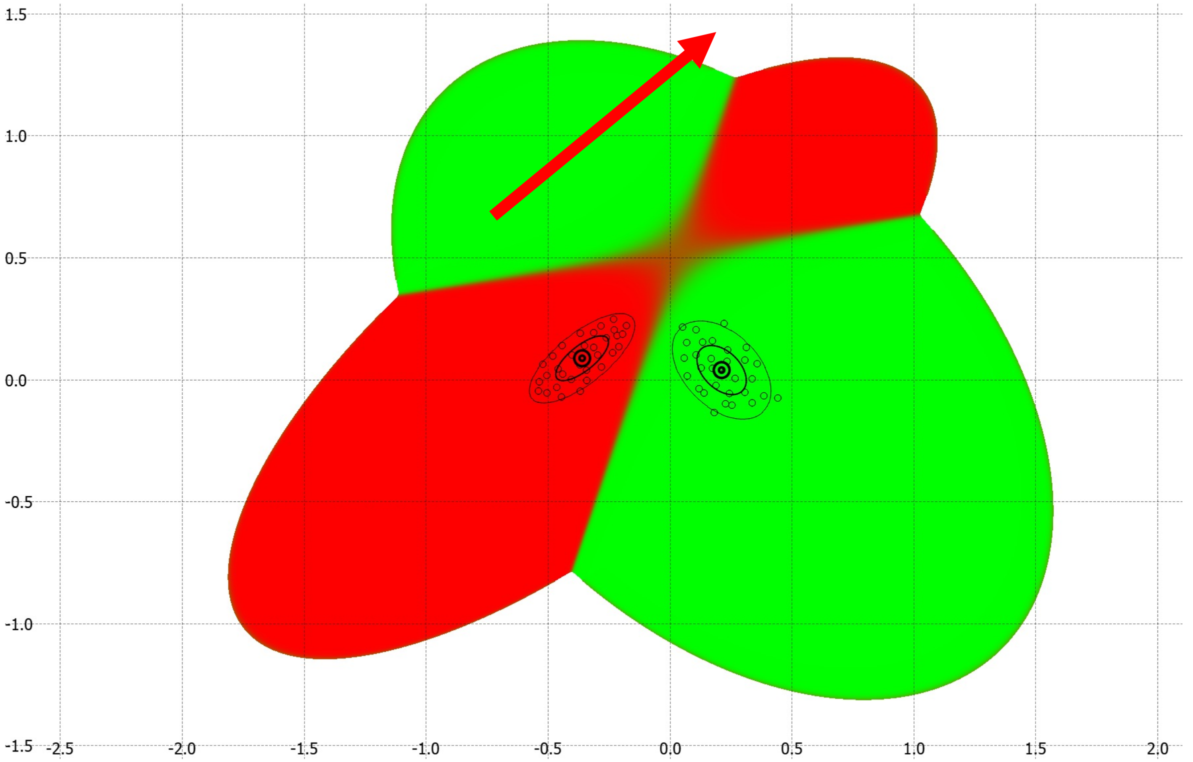
Linear Boundary equidistant from the two centers



Mixture of Gaussians and Bayes’ classification

❖A) c)

- Full type covariance matrices
- Same number of points
- The green class has a larger variance than the red class along its 1^{st} eigenvector. It overcomes the red class far from the data along this direction
- Similarly the red class overcomes the green one along its 1^{st} eigenvector far from the data



Non-linear boundary

Mixture of Gaussians and Bayes' classification

❖B) What happen to the boundary in case Aa) and Ab) if the classes have a much different number of points ?

- Bayes rule:

$$p(y=i|x) = \frac{p(x|y=i)p(y=i)}{p(x)}, \quad i = 1, 2.$$

- Classification boundary is obtained by setting the likelihood ratio to 1

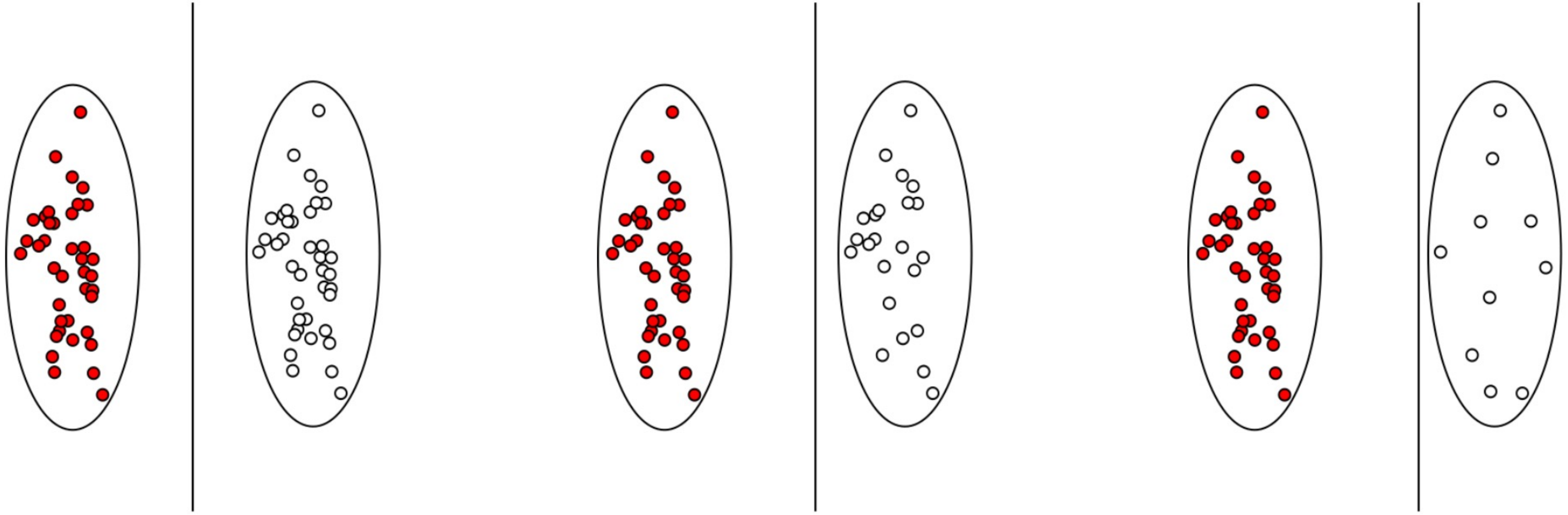
$$\frac{p(y=1|x)}{p(y=2|x)} = \frac{p(x|y=1)p(y=1)}{p(x|y=2)p(y=2)} = \frac{p(x|y=1)}{p(x|y=2)} \times \frac{p(y=1)}{p(y=2)} = 1$$

- Applying the logarithm gives:

$$-(x-\mu^1)\Sigma^{-1}(x-\mu^1)^T + \log p(y=1) = -(x-\mu^2)\Sigma^{-1}(x-\mu^2)^T + \log p(y=2)$$

Mixture of Gaussians and Bayes' classification

❖B) What happen to the boundary in case Aa) and Ab) if the classes have a much different number of points ?



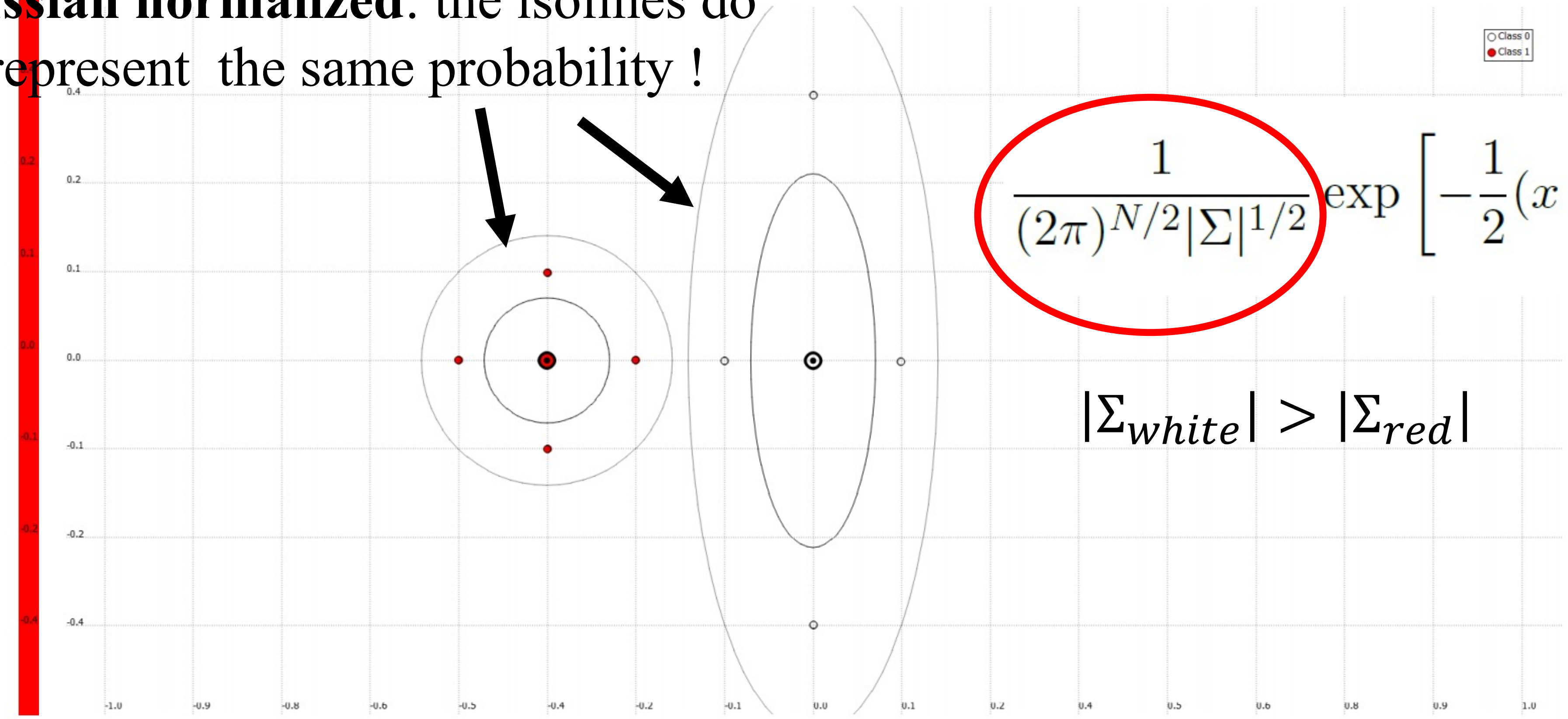
(a) (b) (c)

The boundary is moving based on the ratio between the two classes

Learning Algorithms and Systems Laboratory

❖C) Boundary between normalized spherical and diagonal Gaussian

Gaussian normalized: the isolines do not represent the same probability !

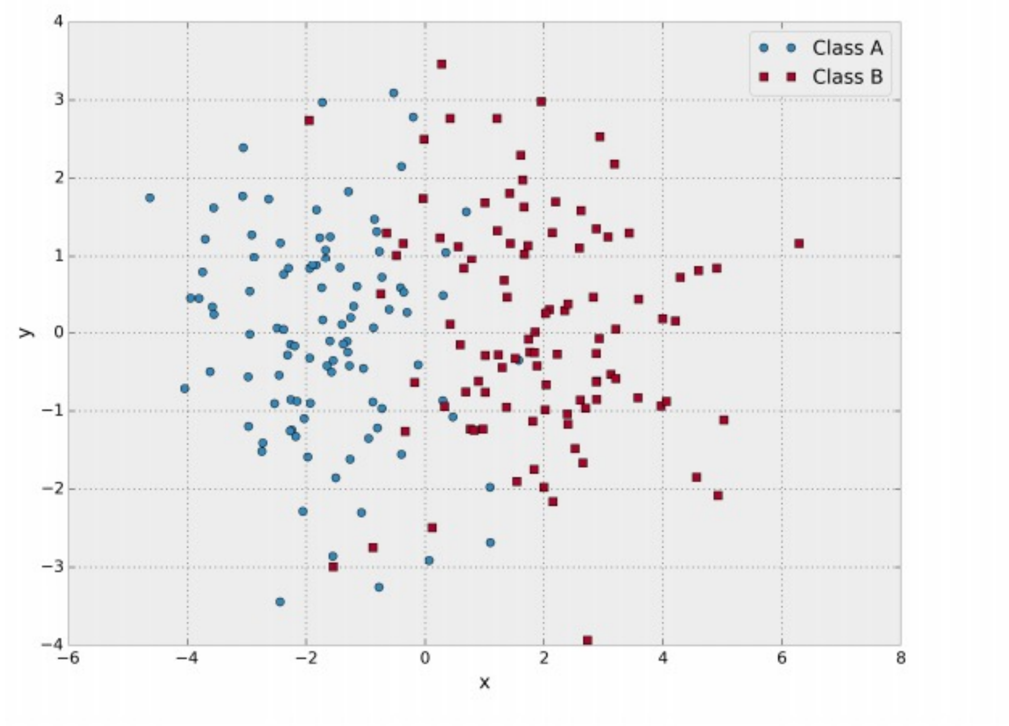


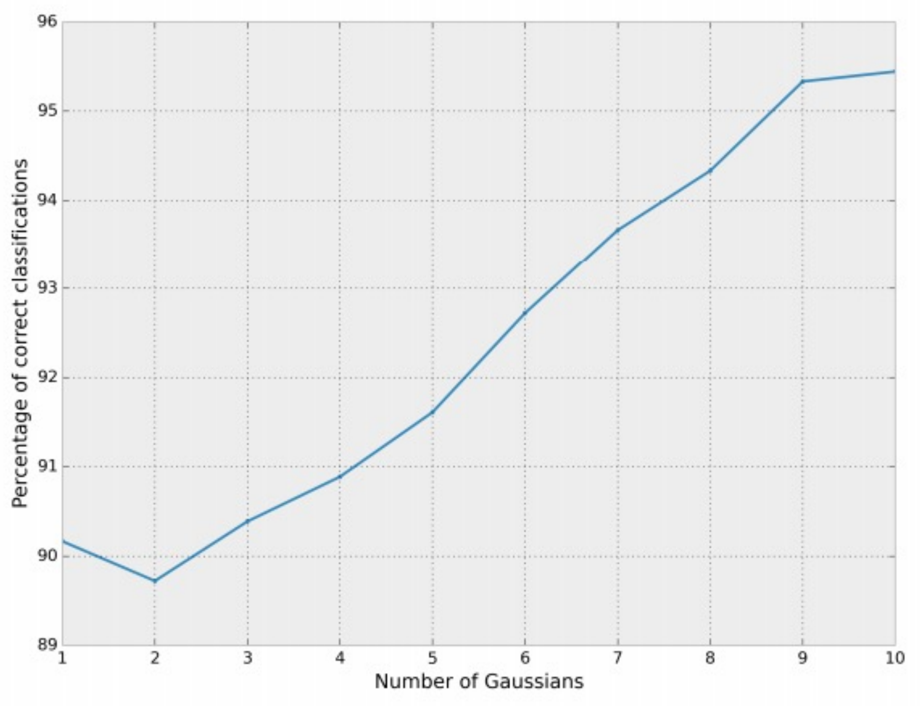
$$\frac{1}{(2\pi)^{N/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right]$$

$$|\Sigma_{white}| > |\Sigma_{red}|$$

Overfitting, Generalization and computational costs

❖A) Joes trains a GMM on each class and evaluates the performance on the **training set**. From the performance obtained he decides to use **10 gaussians**.

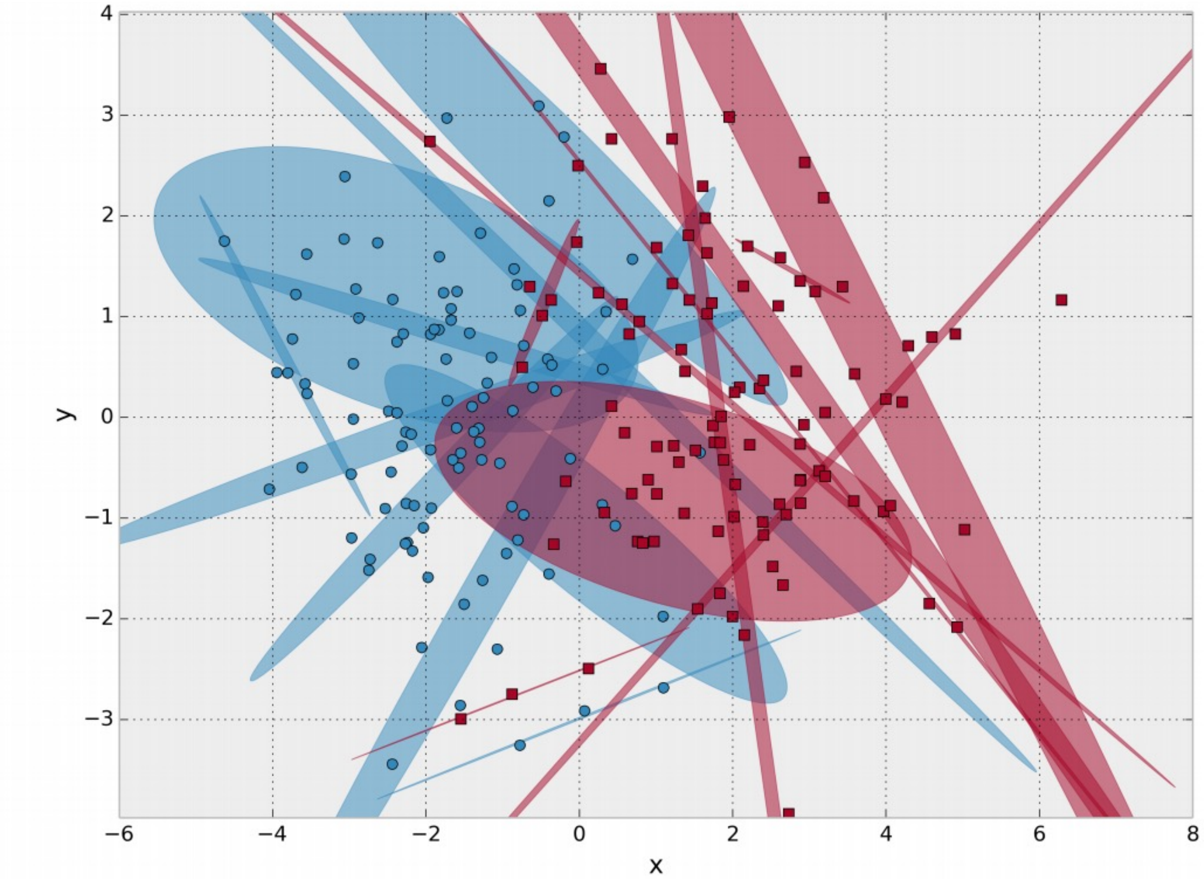




Overfitting, Generalization and computational costs

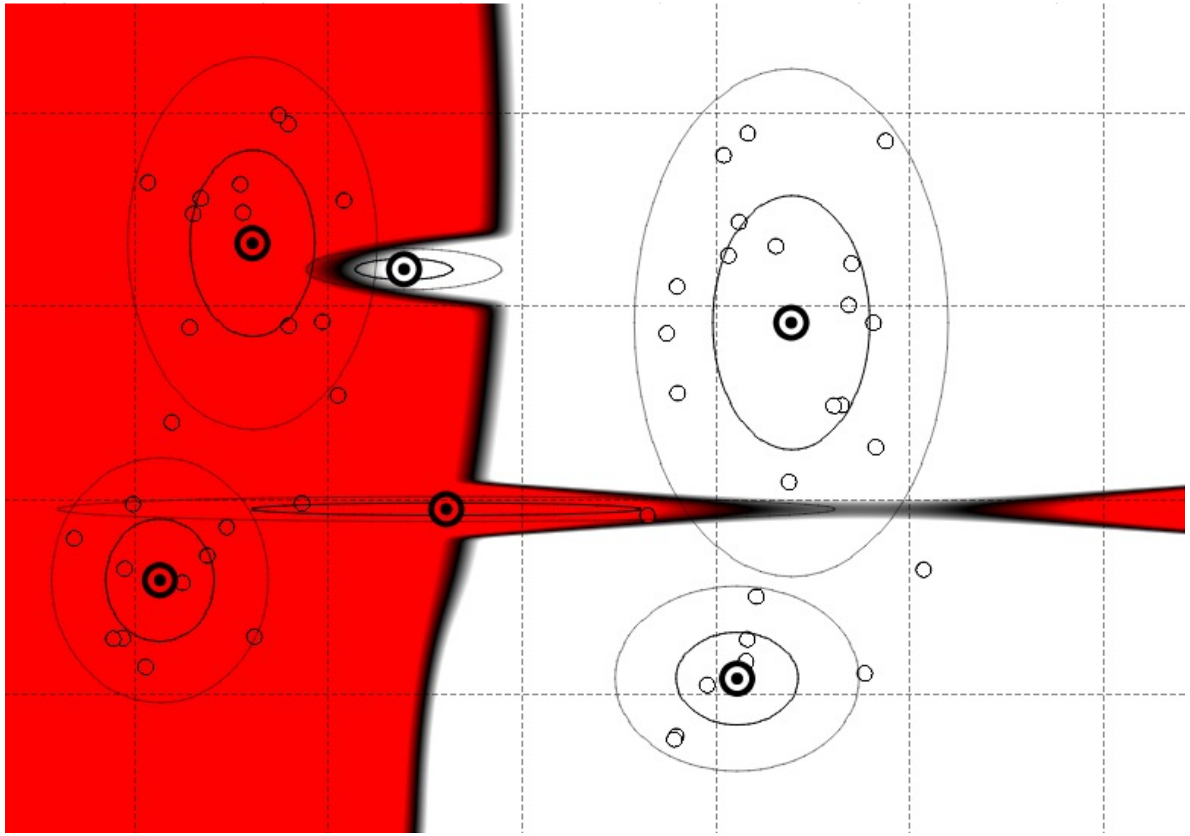
❖A) Using a too high number of gaussians will make the model **overfitting**

- Model should be learned on training set and evaluated on test set: K-fold cross validation
- Selection criterion like BIC can be used to also avoid overfitting



Overfitting, Generalization and computational costs

- B) Example of dataset and choice of GMM leading to overfitting



Overfitting, Generalization and computational costs

❖C) Number of parameters to estimate for GMM

- Full Gaussian: $K\times\Big(\underbrace{N}_{\text{Mean}}+\underbrace{\frac{N(N+1)}{2}}_{\text{Covariance matrix}}\Big)+\underbrace{K-1}_{\text{Weights}}$

K: Number of gaussians
N: Input dimension

- Diagonal Gaussian: $K\times(N+N)+K-1$
- Spherical Gaussian: $K\times(N+1)+K-1$

Overfitting, Generalization and computational costs

Learning Algorithms and Systems Laboratory

❖C) Computational cost per iteration for GMM update step

$$\Sigma_{\text{full}}^{k\;(t+1)} = \frac{\sum_j p(k|x^j, \Theta^{(t)})(x^j - \mu^{k\,(t+1)})(x^j - \mu^{k\,(t+1)})^T}{\sum_j p(k|x^j, \Theta^{(t)})} \quad \rightarrow O(KN^2M)$$

$$\Sigma_{\text{dia}}^{k\;(t+1)} = \text{diag}((\sigma_1^{k\,(t+1)})^2, \ldots, (\sigma_N^{k\,(t+1)})^2) \quad \rightarrow O(KNM)$$

$$\Sigma_{\text{iso}}^{k\;(t+1)} = \text{diag}((\sigma^{k\,(t+1)})^2) \quad \rightarrow O(KM)$$

where

$$(\sigma_i^{k\,(t+1)})^2 = \frac{\sum_j p(k|x^j, \Theta^{(t)})(x_i^j - \mu_i^{k\,(t+1)})^2}{\sum_j p(k|x^j, \Theta^{(t)})}, \quad \forall i = 1..N$$

are the variances along each dimension and

$$(\sigma^{k\,(t+1)})^2 = \frac{\sum_j p(k|x^j, \Theta^{(t)})\|x^j - \mu^{k\,(t+1)}\|^2}{N\sum_j p(k|x^j, \Theta^{(t)})}$$

is the variance averaged over all dimensions (isotropic).

K: Number of gaussians
N: Input dimension
M: Number of points