Learning from Data: Density Estimation - Gaussian Mixture Models

Amos Storkey, School of Informatics
University of Edinburgh

Semester 1, 2004

Summary

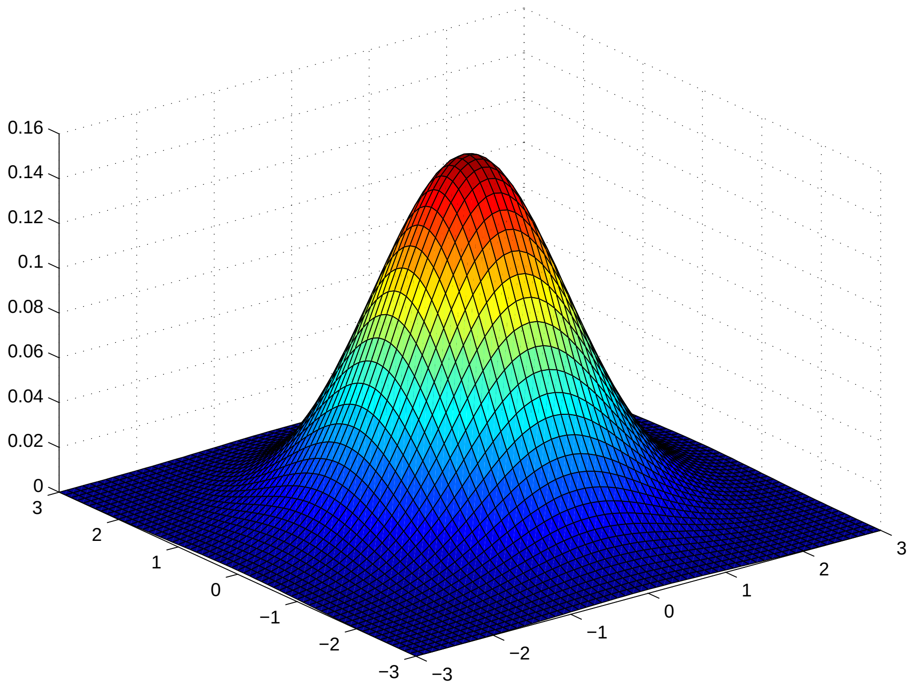
- Class conditional Gaussians
- What about if the class is unknown.
- What if just have a complicated density
- Can we learn a clustering?
- Can we learn a density?

Gaussian: Reminder

- The vector $\mathbf{x}$ is multivariate Gaussian if for mean $\boldsymbol{\mu}$ and covariance matrix Σ, it is distributed according to

$$P(\mathbf{x}|\boldsymbol{\mu},\Sigma) = \frac{1}{|2\pi\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

Multivariate Gaussian: Picture



Reminder: Class Conditional Classification

- Have real valued multivariate data, along with class label for each point.
- Want to predict the value of the class label given some new point.
- Presume that if we take all the points with a particular label, then we believe they were sampled from a Gaussian.
- How should we predict the class at a new point?

Reminder: Class Conditional Classification

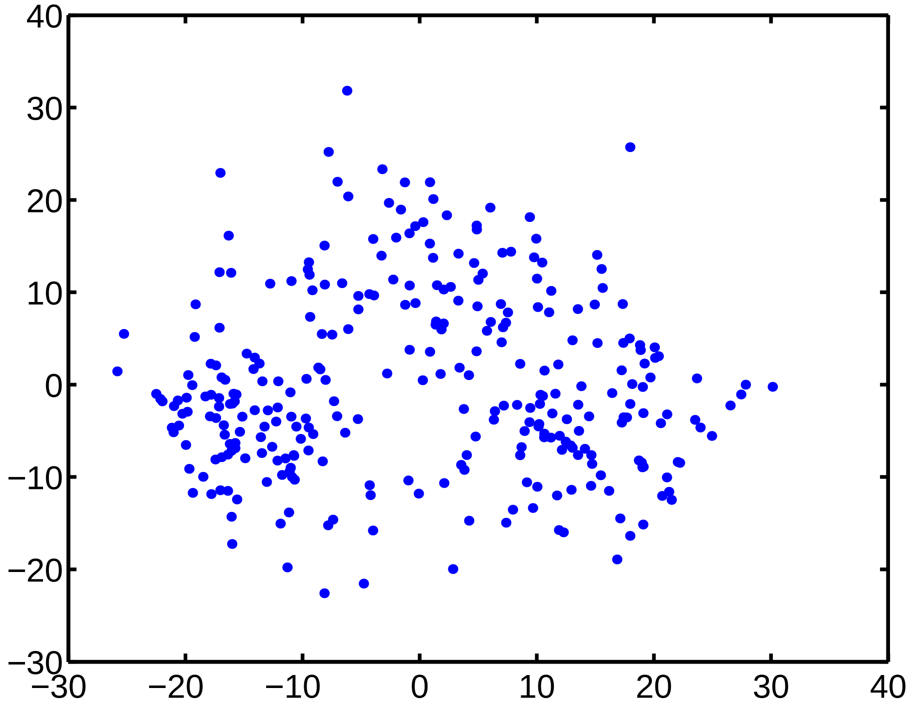
- Learning: Fit Gaussian to data in each class (class conditional fitting). Gives $P(\text{position}|\text{class})$
- Find estimate for probability of each class $P(\text{class})$
- Inference: Given a new position, we can ask "What is the probability of this point being generated by each of the Gaussians."
- Pick the largest and give probability using Bayes rule

$$P(\text{class}|\text{position}) \propto P(\text{position}|\text{class})P(\text{class})$$

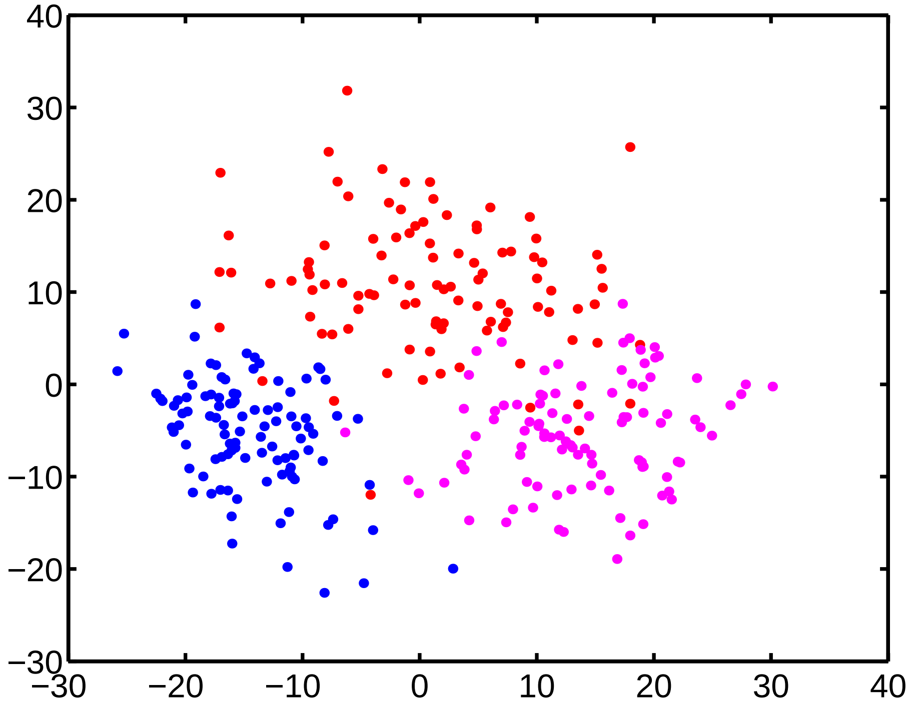
Unsupervised Problem

- Given exactly the same problem, but without any class labels, can we still solve it?
- Presume we know the number of classes and know they are Gaussian.
- Can we model the underlying distribution?
- Can we cluster the data into different classes?
- Effectively we want to allocate a class label to each point.

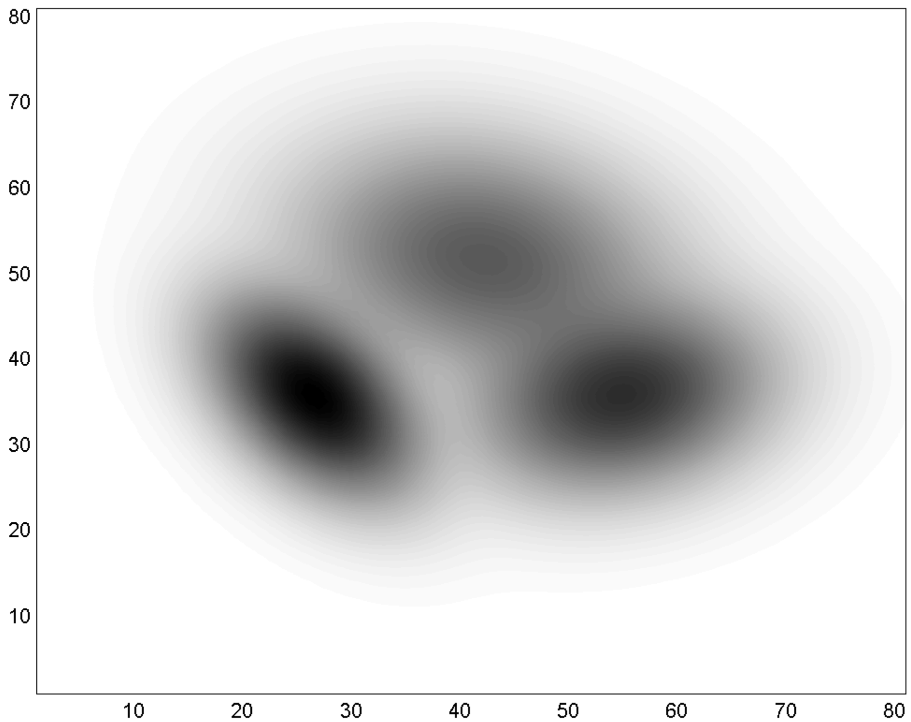
Example



Example



Example



Can solve either-or

- If we know which clusters the points belong to we can solve (learning a class-conditional model).

- If we know what the Gaussian clusters are we can solve (inferential classification using a class-conditional model)

- The first case was just what we did when we had training data.

- The second was just what we did using Bayes rule for new test data.

- But how can we do the two together?

Iterate?

- Could just iterate: Guess the cluster values. Calculate parameters. Find maximum cluster values.
- No reason to believe this will converge.
- Problem is that we have probability of belonging to a cluster, but we allocate all-or nothing.

Use Gradient Optimisation

- Write out model. Calculate derivatives, optimise directly using conjugate gradients.
- Will work. Quite complicated. Not necessarily fastest approach.

Probabilistic Allocation

- Stronger: if we know a *probabilistic* allocation to clusters we can find the parameters of the Gaussians.

- If we know the parameters of the Gaussians we can do a probabilistic allocation to clusters.

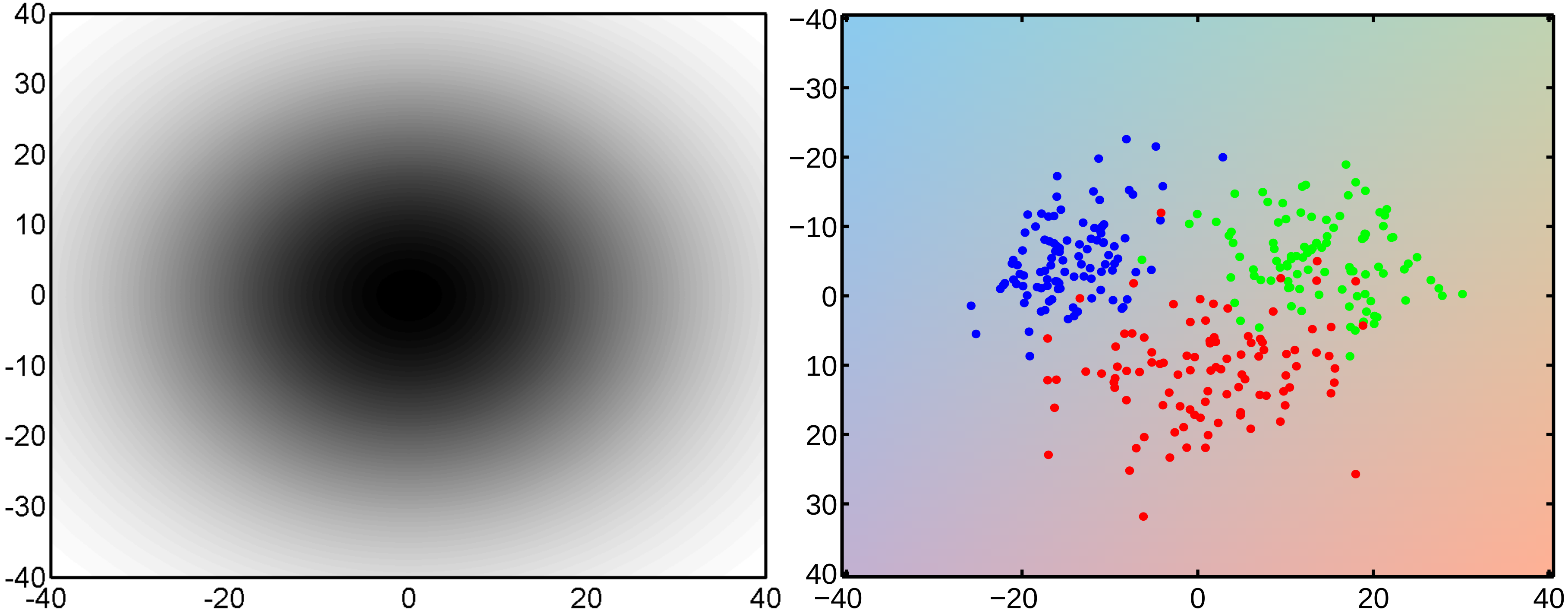
- Convergence guarantee!

- Convergence-to-a-local-maximum-likelihood-value guarantee!!

EM Algorithm

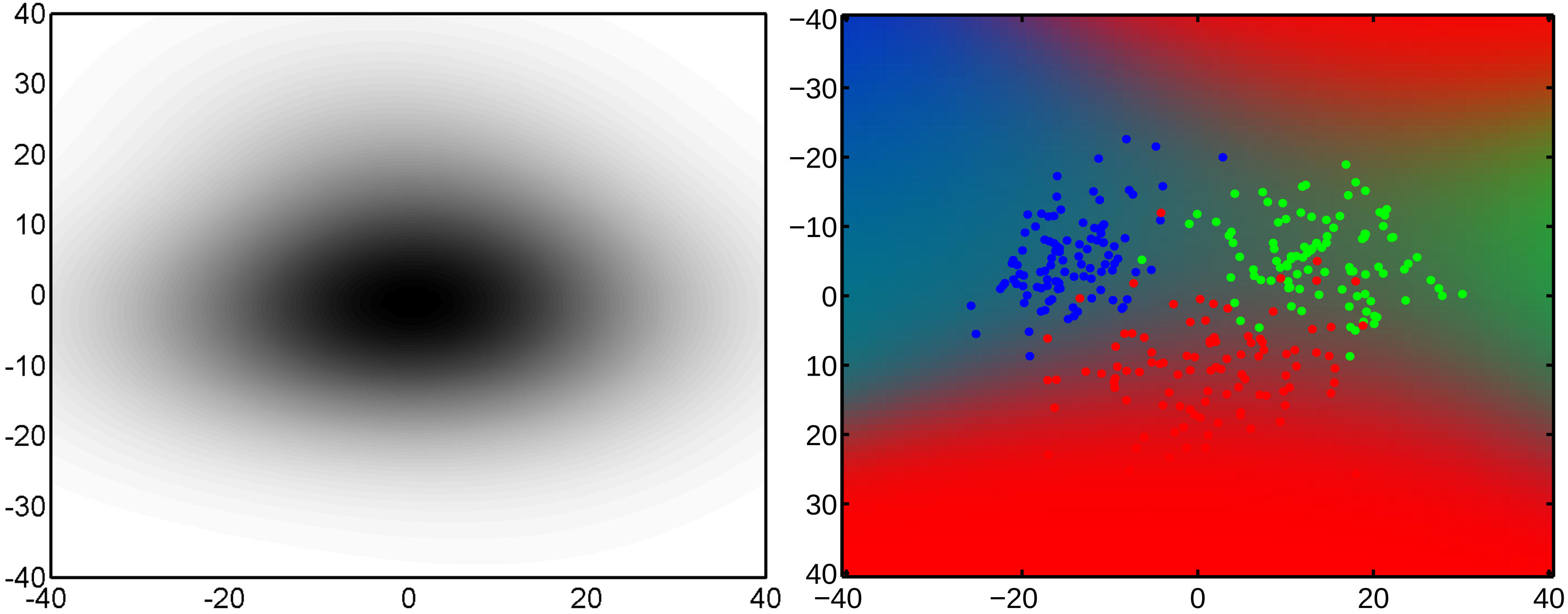
- Choose number of mixtures.
- Initialise Gaussians and mixing proportions.
- Calculate *responsibilities*:
- $P(i|\mathbf{x}^\mu)$ - the probability of data point $\mathbf{x}^\mu$ belonging to cluster i given the current parameter values of the Gaussians and mixing proportions.
- Pretend the responsibilities are the truth. Update the parameters using a maximum likelihood method (generalisation of class conditional learning).
- But the responsibilities are not the truth, so update them and repeat.
- Will converge to maximum likelihood parameter values.

Example



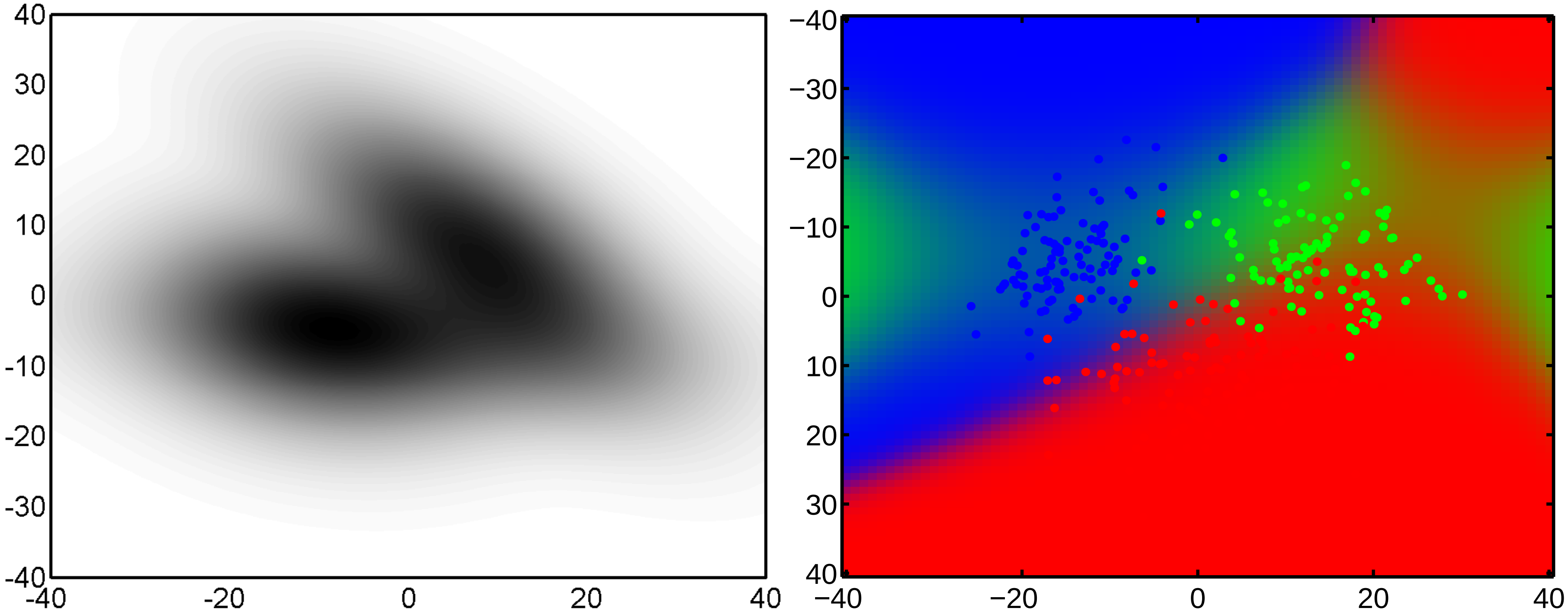
40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40
−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Example



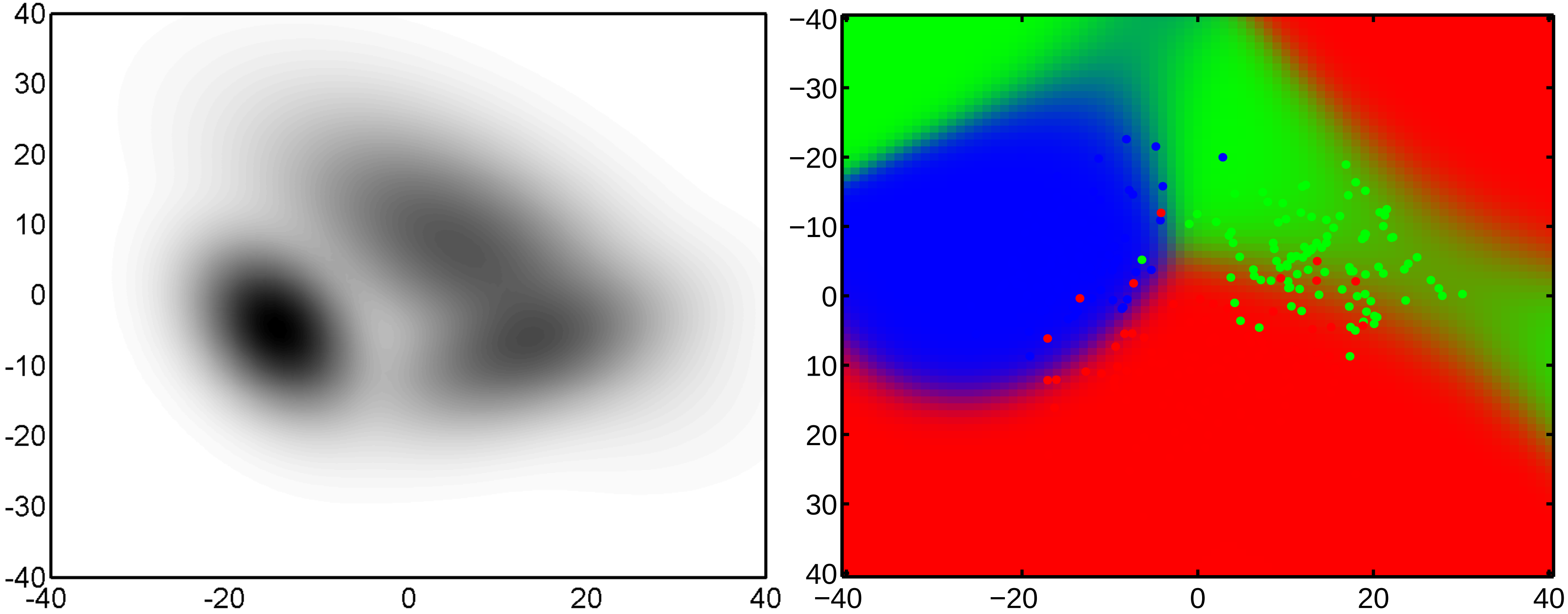
40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40
−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Example



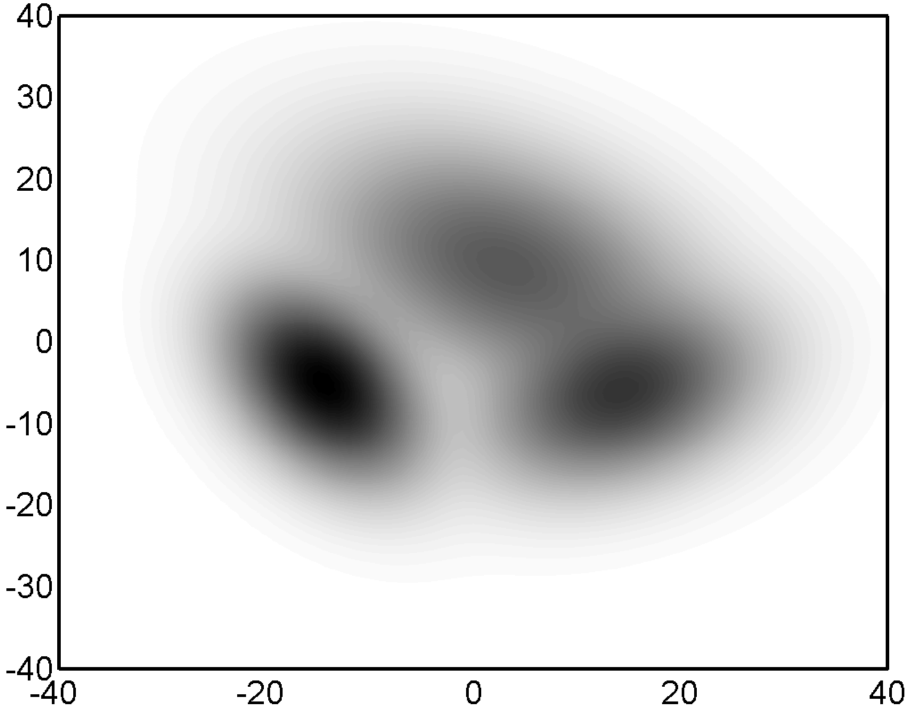
40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40
−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Example

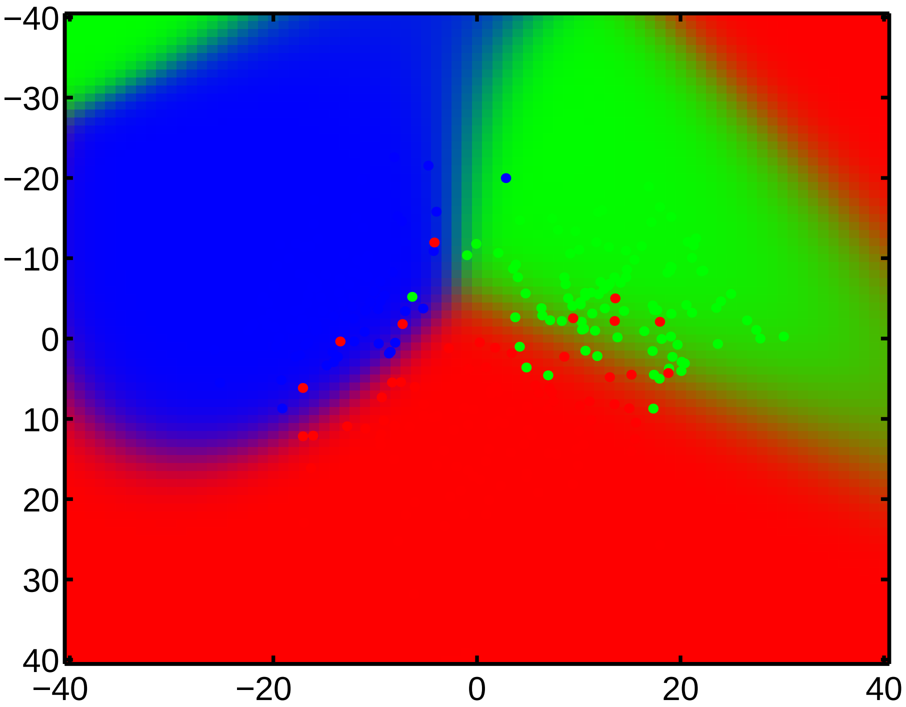


40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40
−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Example

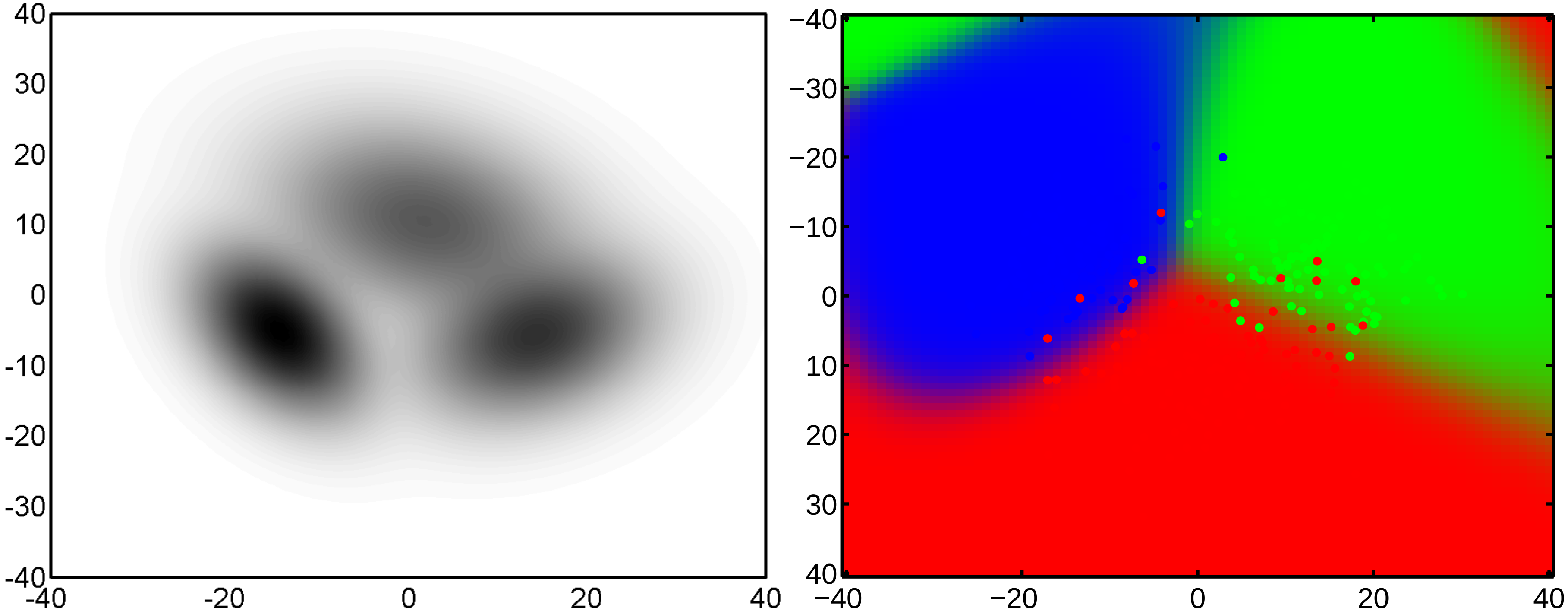


40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40



−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Example



40
30
20
10
0
-10
-20
-30
-40
-40
-20
0
20
40
−40
−30
−20
−10
0
10
20
30
40
−40
−20
0
20
40

Labelling

- The labels are permutable.
- We can use partially labelled data to set the cluster labels: fix responsibilities deterministically.
- EM algorithm for the rest.

Choice of number of mixtures

- How good is the likelihood?
- More mixtures, better likelihood.
- One mixture on each data point: infinite likelihood.
- Need regularisation on Gaussian widths (i.e. covariances).
- Aside: Bayesian methods account for probability mass in parameter space.

Initialisation

- K-Means
- Uses K-NN for clustering.
- See lecture notes.

Inference and Clustering

- Just as with class conditional Gaussian model
- Have mixture parameters.
- Calculate posterior probability of belonging to a particular mixture.

Covariances

- Full covariances
- Diagonal covariances
- Others types (factor analysis covariances - bit like PCA).

Summary

- Mixture Models - class conditional models without the classes!
- EM algorithm
- Using class conditional models
- Issues: number of mixtures, covariance types etc.