1. Classification and Maximum-Likelihood Estimation

Fred Sigworth
Cellular and Molecular Physiology, Yale University

Clustering in 3D

The problem of detecting heterogeneity in 2D

Maximum Likelihood

Two simple examples

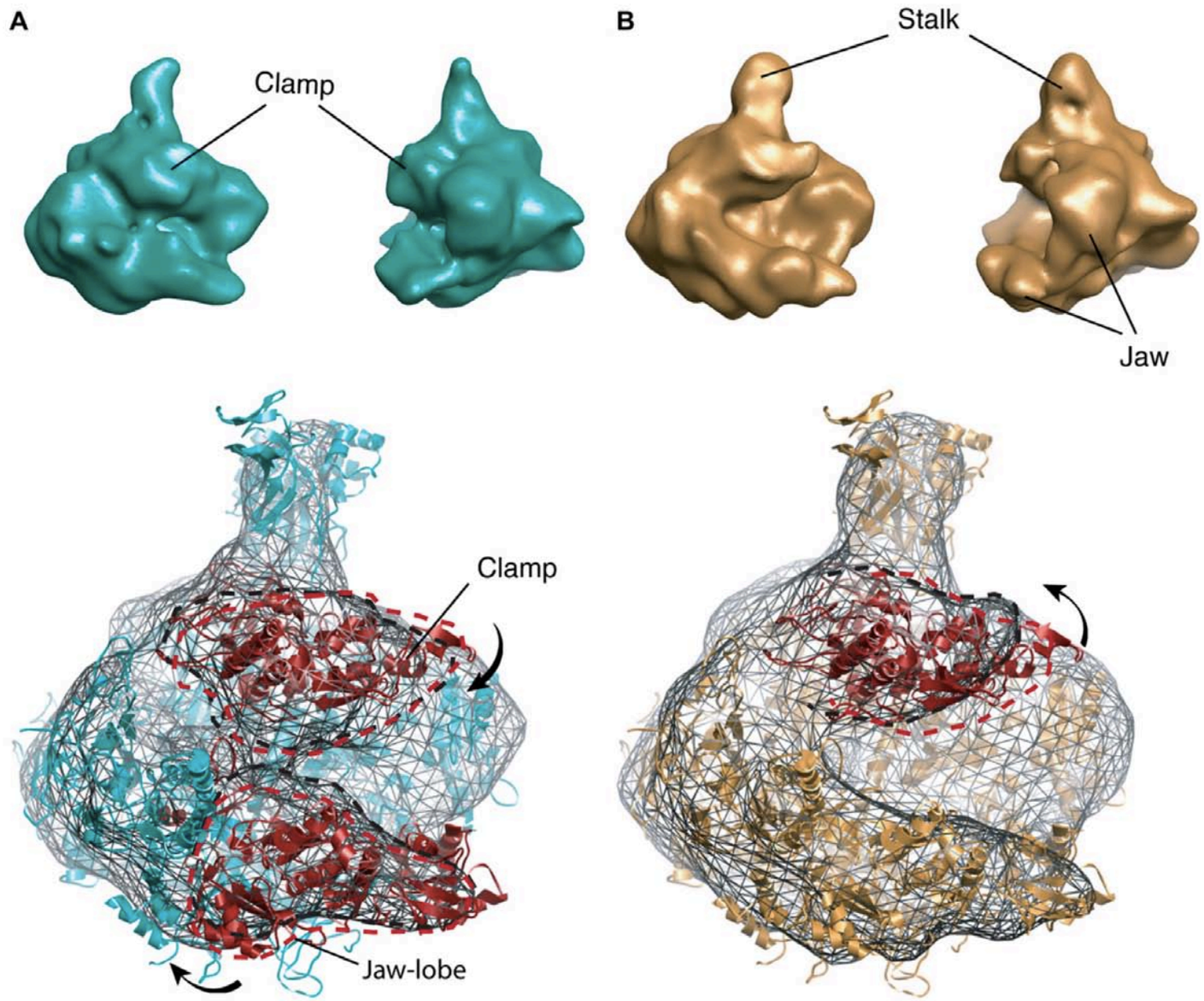
Hidden variables and the EM algorithm

An example of heterogeneity

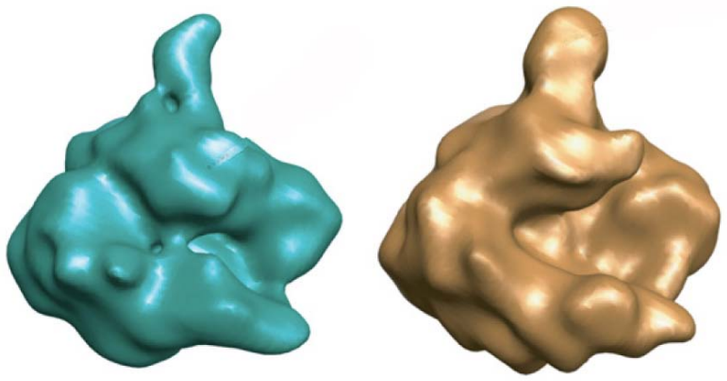
Structure *14*, 1691–1700, November 2006 ©2006 Elsevier Ltd All rights reserved DOI 10.1016/j.str.2006.09.011

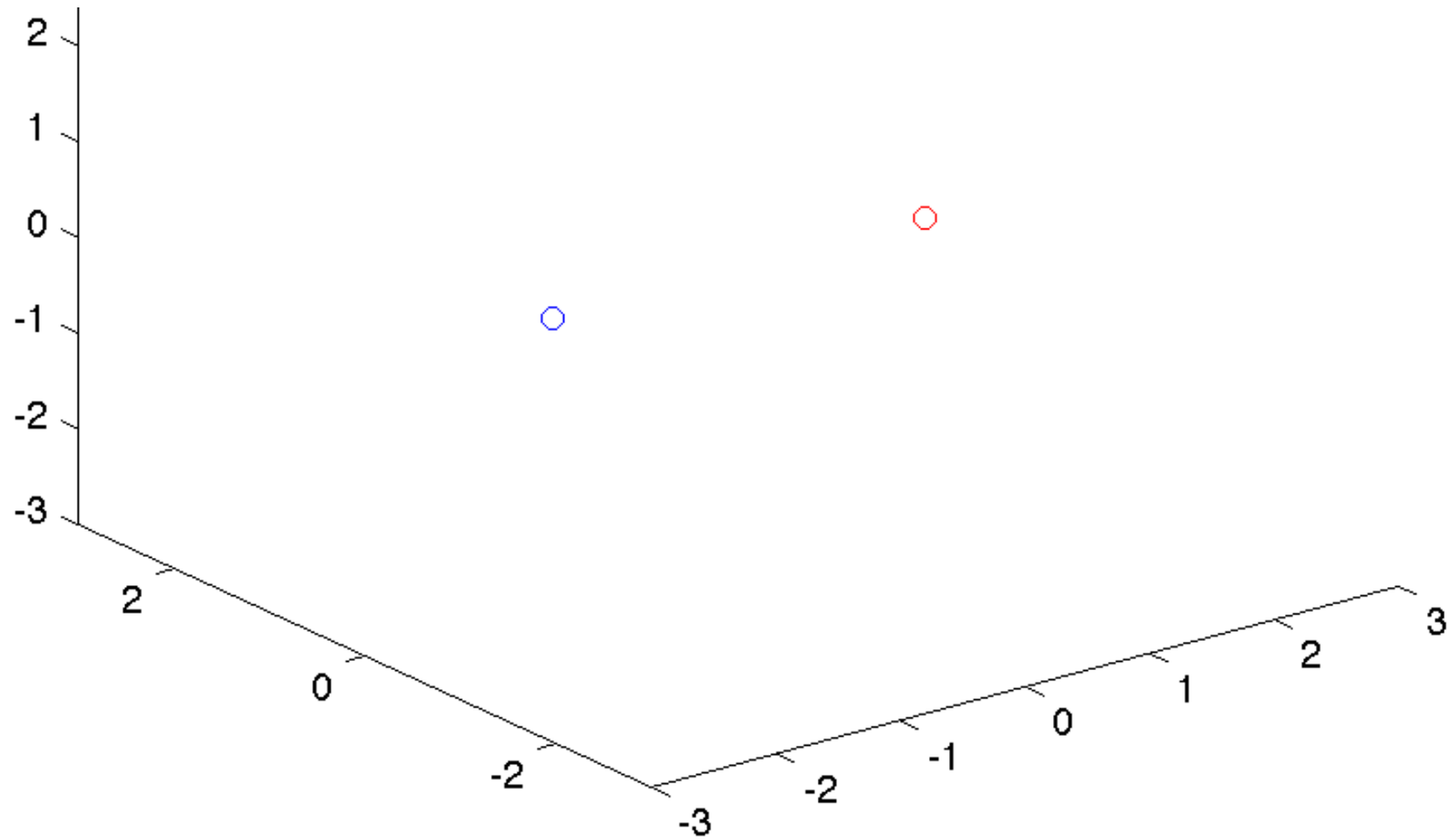
Molecular Architecture and Conformational Flexibility of Human RNA Polymerase II

Seth A. Kostek,[1,3,4] Patricia Grob,[1,4] Sacha De Carlo,[2] J. Slaton Lipscomb,[1,2] Florian Garczarek,[3] and Eva Nogales[1,2,3,*]

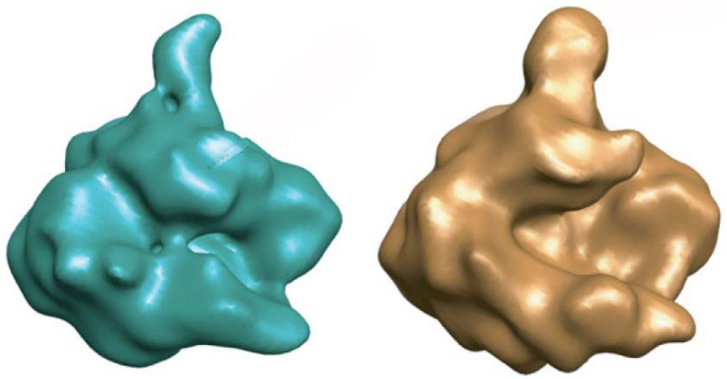


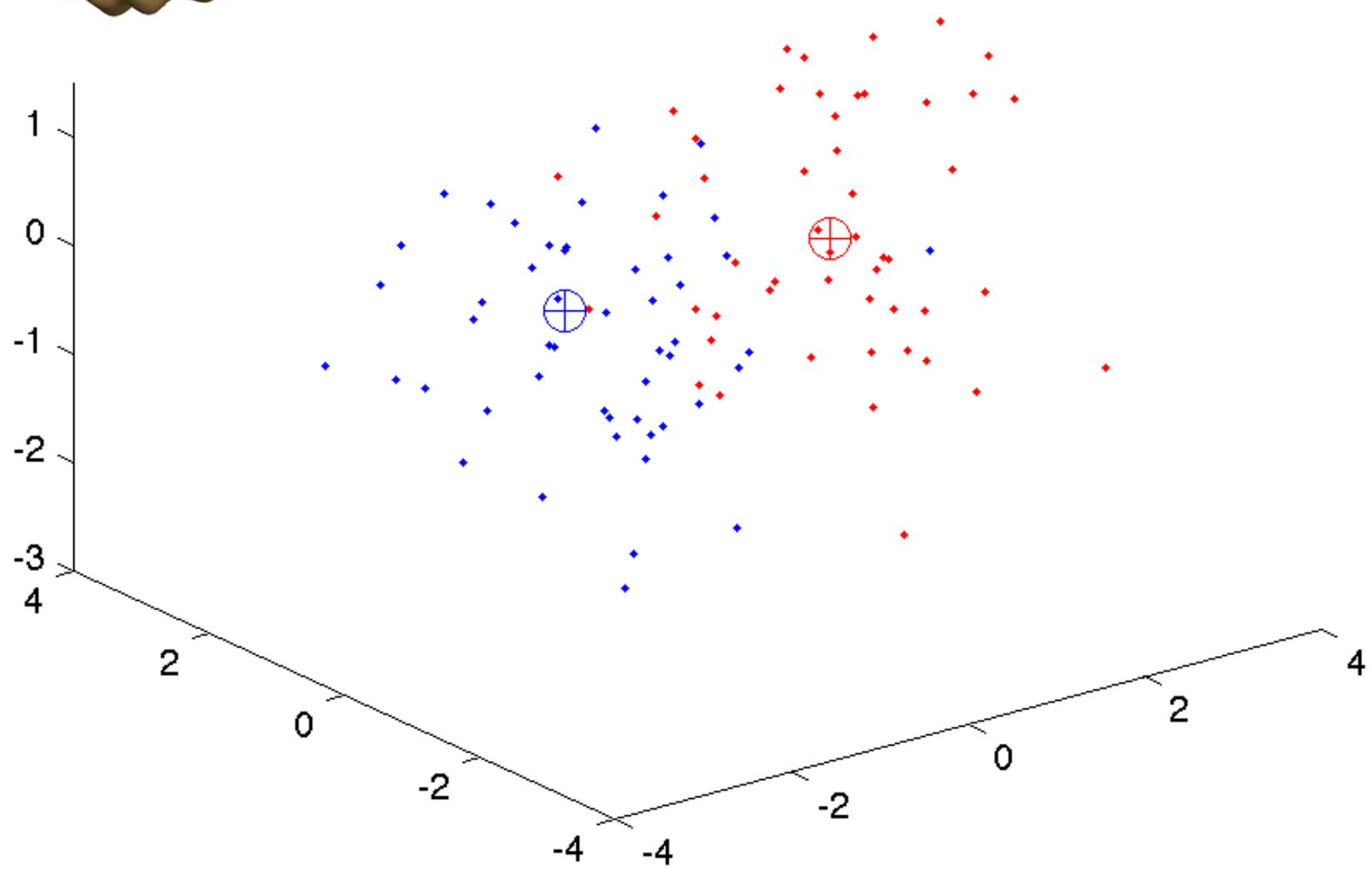
Representation of 3D models in $\mathbb{R}^M$



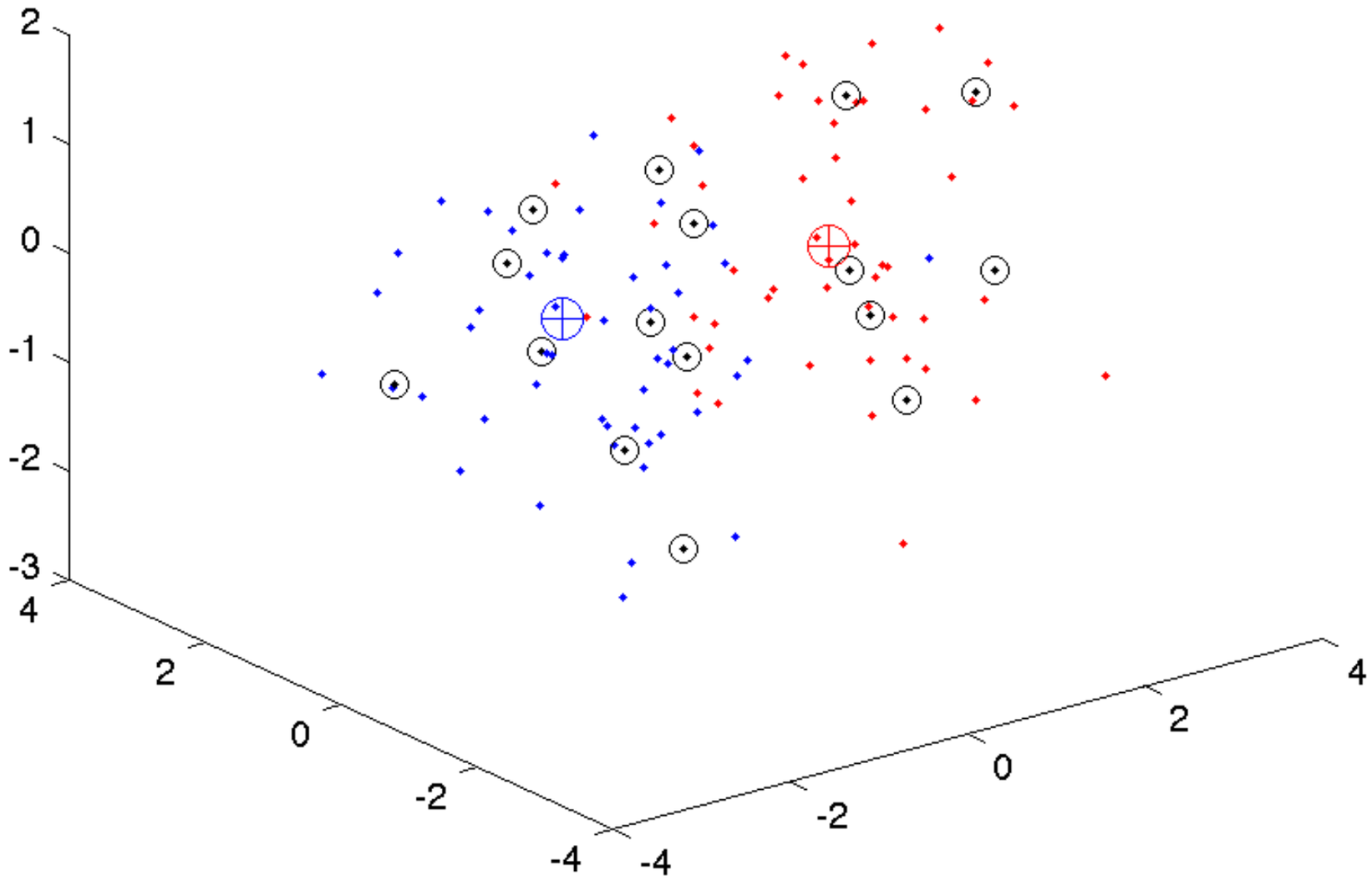


Representation of 3D models in $\mathbb{R}^M$

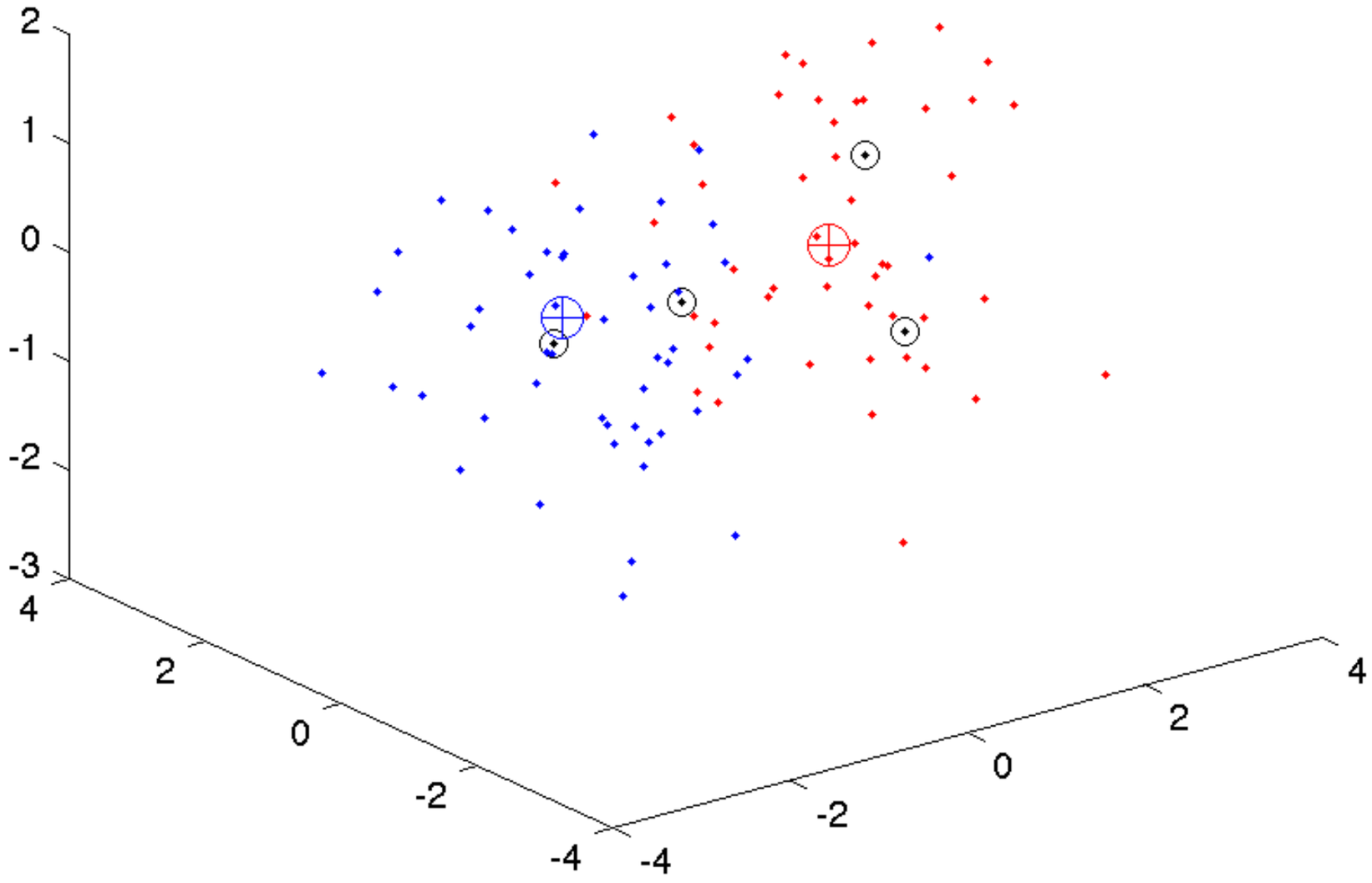




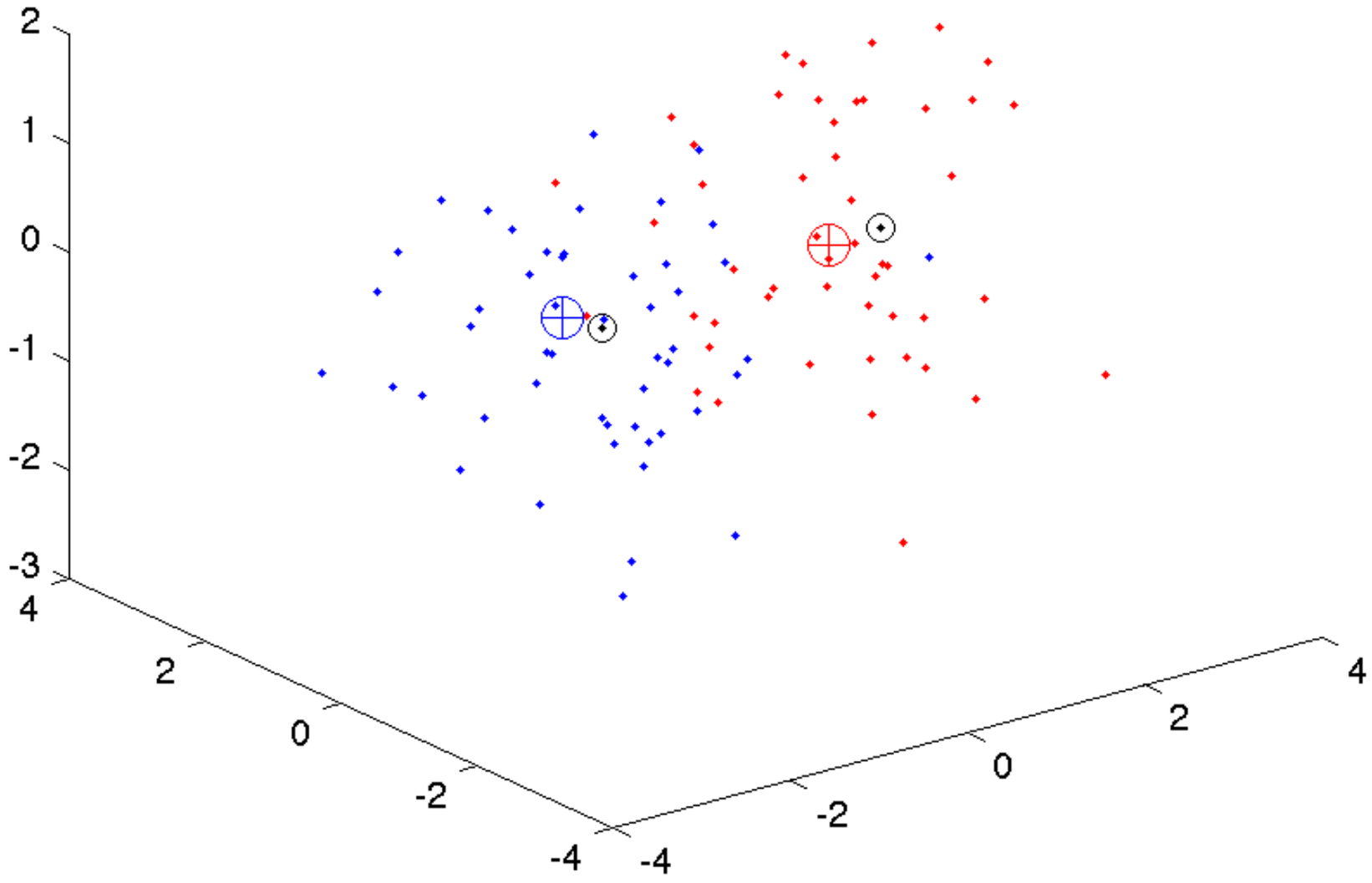
Clustering



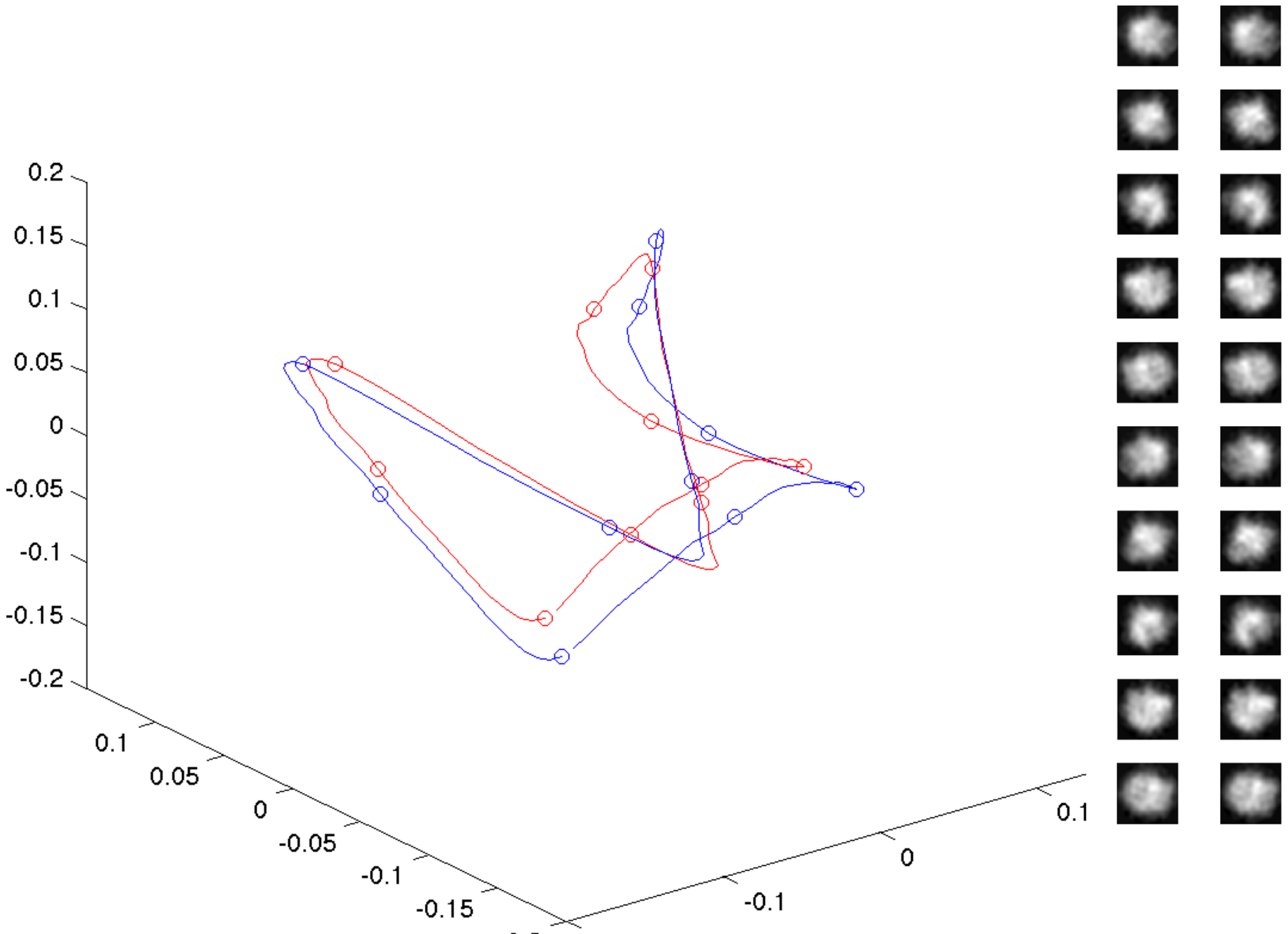
Clustering



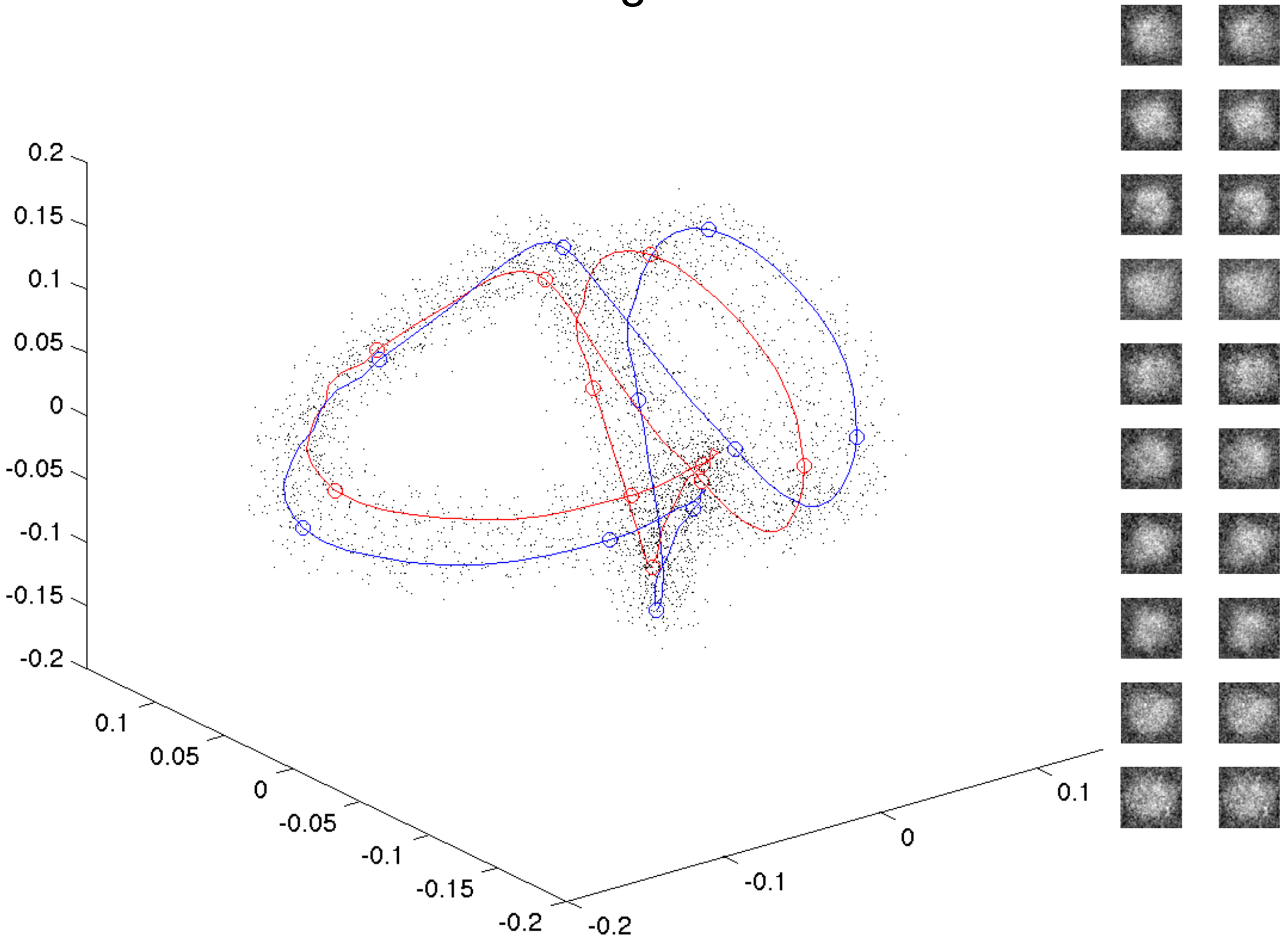
Clustering



Representation of 2D projections in $\mathbb{R}^N$



In the presence of noise, it’s hard to sort the particle images.



Introduction to maximum-likelihood methods

When we do a single-particle reconstruction, what is the quantity that we are maximizing?

Conventional align-and-average methods maximize the power in the reconstructed volume.

ML methods maximize a statistical quantity that is not rigorously a probability, so it's called the likelihood.

- Let Θ be a description of the model, i.e. the density maps of the reconstructions.
- Let $\mathbf{X}$ be the data, that is the collection of acquired images.
- Then $P(\Theta \mid \mathbf{X})$ would be the probability of the model Θ being the correct one.

Introduction to maximum-likelihood methods

Given a stack of images $\mathbf{X} = \{X_1, X_2, ..., X_N\}$ find the best "model", that is the set of reconstructions and other parameters

$$\Theta = \{V_1, V_2, ..., V_M, a_1, a_2, ..., , a_m, \sigma\}$$

What criterion should be used for the "best?"

How about maximizing the probability of the reconstructions given the data,

$$P(\Theta \mid \mathbf{X})$$

$P(\Theta \mid \mathbf{X})$ is difficult to compute...or define...

However, we can compute $P(\mathbf{X} \mid \Theta)$. Let's define the likelihood as a function of Θ

$$\text{Lik}(\Theta) = P(X \mid \Theta)$$

MLE and MAP Estimation

The probability of the model is related to the likelihood by Bayes' theorem,

$$p(\Theta \mid \mathbf{x}) = p(\mathbf{x} \mid \Theta)\frac{p(\Theta)}{p(\mathbf{x})}$$

The maximum-likelihood estimate (MLE) optimizes $p(\mathbf{x} \mid \Theta)$.

$$\text{Experiment} \longrightarrow \Theta$$

The maximum a posteriori estimate (MAP) optimizes $p(\mathbf{x} \mid \Theta)p(\Theta)$.

$$p(\Theta) \longrightarrow \text{Experiment} \longrightarrow \Theta$$

a priori *a posteriori*

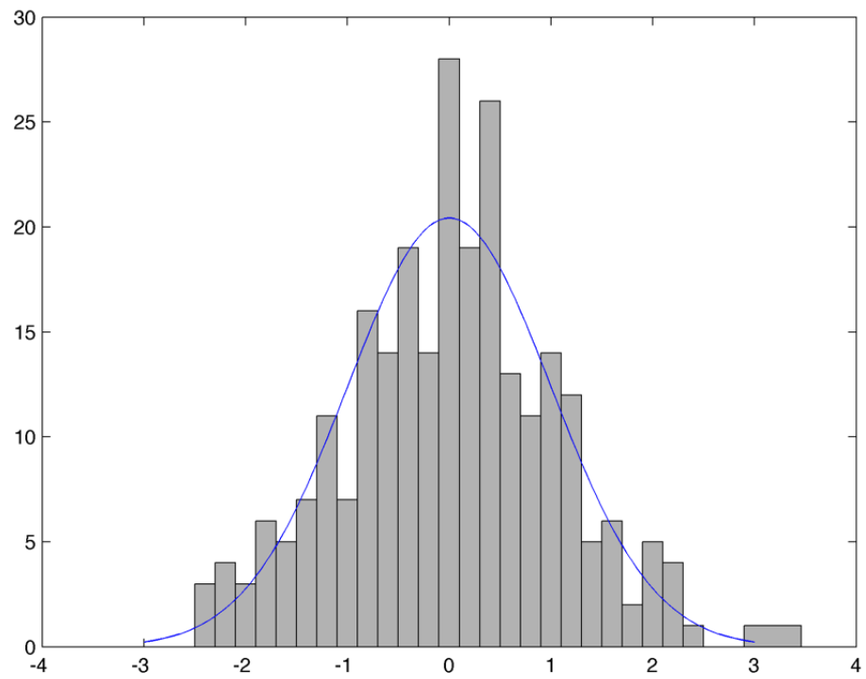
MLE Example 1: Gaussian random numbers

$$\text{Lik} = \varepsilon f(x_1) \cdot \varepsilon f(x_2) \cdot \ldots \cdot \varepsilon f(x_N)$$

where ε is the measurement resolution and the pdf is

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2} \right).$$

x
0.285
0.826
-0.008
0.858
0.775
1.306
1.232
0.959
-1.655
-0.990



MLE Example 1: Gaussian random numbers

To maximize L we set the derivatives to zero,

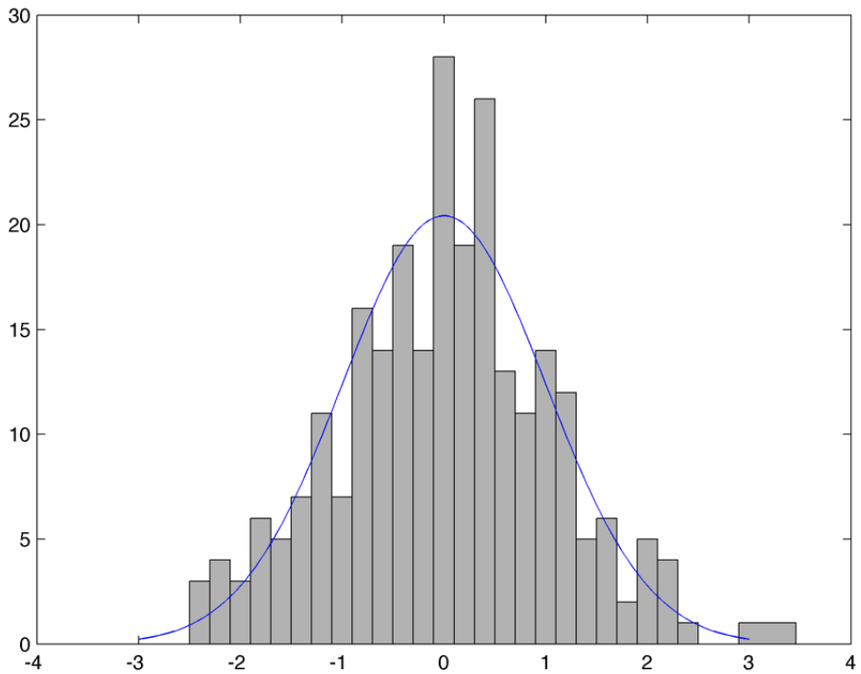
$$L = N\ln(\varepsilon) - \frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}(x_i - \mu)^2$$

$$\frac{\partial L}{\partial \mu} = 0 \quad \Rightarrow \quad \hat{\mu} = \frac{1}{\mathrm{N}}\sum_{i=1}^{N} x_i$$

$$\frac{\partial L}{\partial \sigma} = 0 \quad \Rightarrow \quad \sigma^2 = \frac{1}{\mathrm{N}}\sum_{i=1}^{N}(x_i - \mu)^2$$

In this case the MLE is equal to the least-squares estimate.

x
0.285
0.826
-0.008
0.858
0.775
1.306
1.232
0.959
-1.655
-0.990



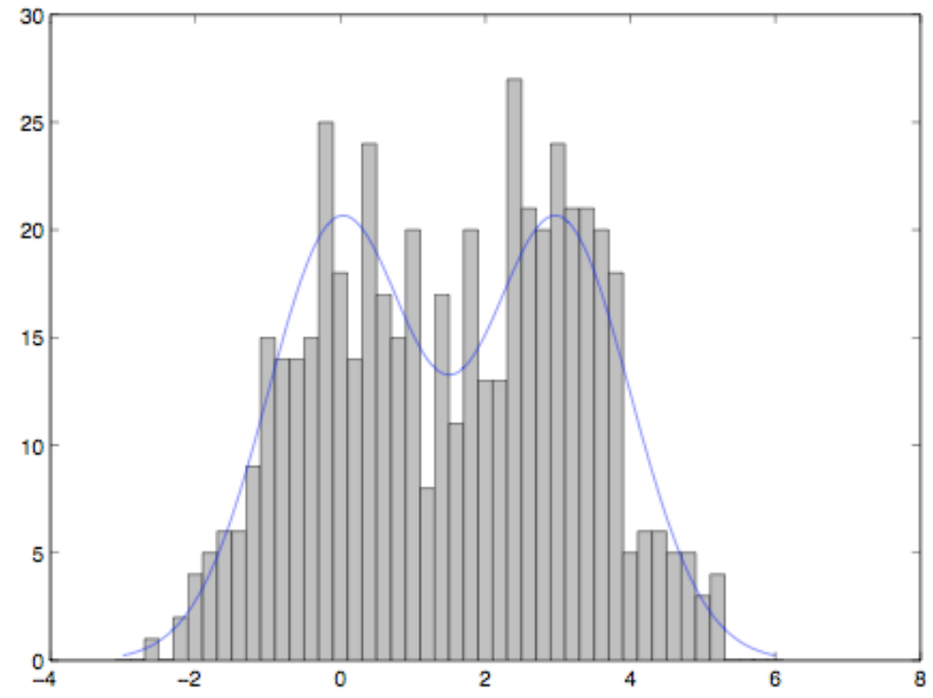
Example 2: Mixture of Gaussians

$$f(x) = a_1 f_1(x) + a_2 f_2(x), \qquad f_j(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x-\mu_1)^2}{2\sigma^2}\right)$$

$$L = \frac{-N}{2}\ln(2\pi\sigma^2) - \sum_{i=1}^{N} \ln\left[a_1 \exp\left(\frac{-(x-\mu_1)^2}{2\sigma^2}\right) + a_2 \exp\left(\frac{-(x-\mu_2)^2}{2\sigma^2}\right)\right]$$

Taking the derivatives of L is not going to be easy. How to maximize it?

x
2.9699
3.0242
-0.7529
3.0639
0.4348
1.3791
0.6150
3.1327
2.9938
1.1833



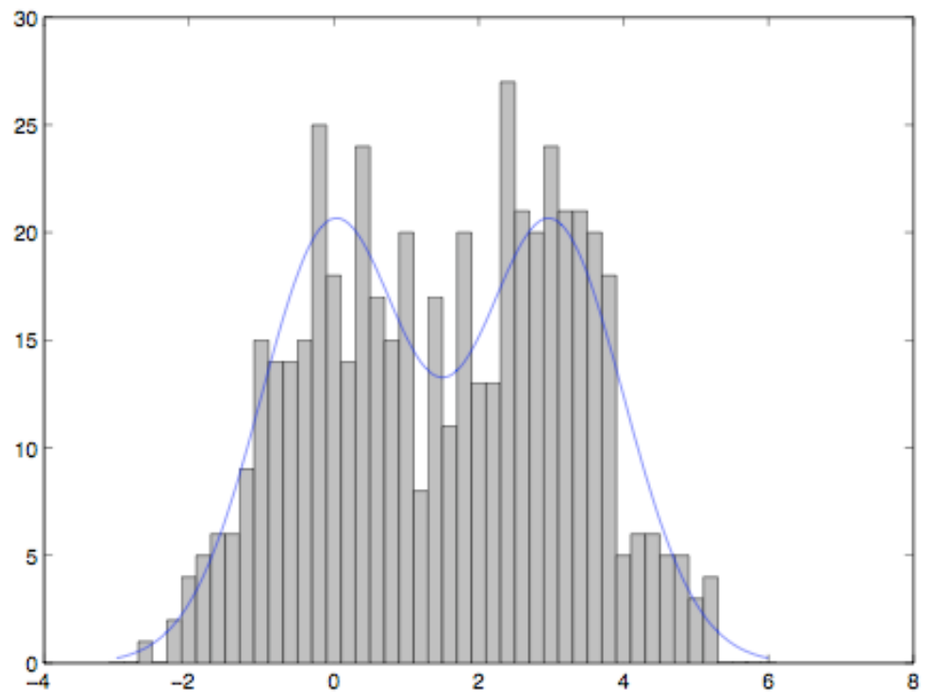
Example 2: Mixture of Gaussians

Suppose we had extra information in the form of "switch variables" z_i. Then estimating the two mean values and the weights would be really easy:

$$\hat{\mu}_1 = \frac{\sum_i z_i x_i}{\sum_i z_i} \qquad \hat{\mu}_2 = \frac{\sum_i (1-z_i) x_i}{\sum_i (1-z_i)}$$

$$a_1 = \frac{1}{N}\sum_i z_i \qquad a_2 = \frac{1}{N}\sum_i (1-z_i)$$

X	Z
0.9863	0
2.9980	1
1.8384	0
0.2488	0
1.6752	0
2.6736	1
2.0572	1
2.6596	1
3.6584	1
2.4223	1



Example 2: Mixture of Gaussians

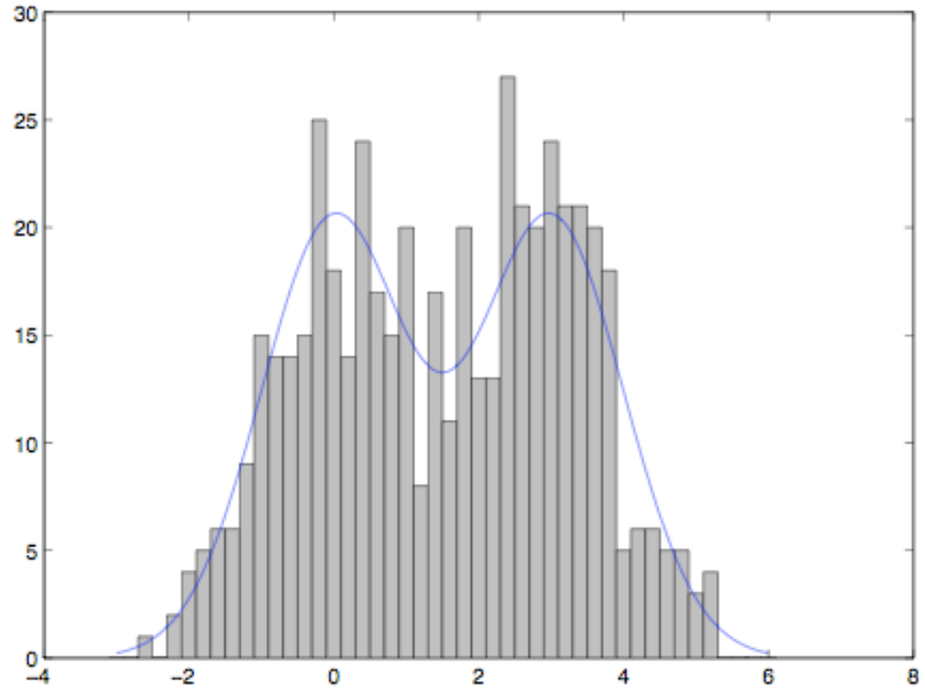
The EM algorithm

Given estimates of the unknown model variables μ_1, μ_2, a_1 and a_2, compute expectation values of the z's:

$$\hat{z}_i = \frac{a_1 f_1(x_i)}{a_1 f_1(x_i) + a_2 f_2(x_i)} \qquad \hat{z}_2 = \frac{a_2 f_2(x_i)}{a_1 f_1(x_i) + a_2 f_2(x_i)}$$

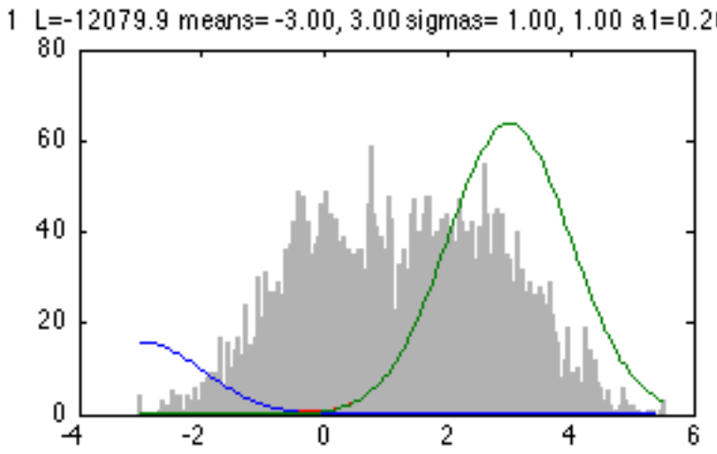
Use these instead of the true z's!

X	Z	1-Z'
0.9863	0	0.8236
2.9980	1	0.0111
1.8384	0	0.2660
0.2488	0	0.9771
1.6752	0	0.3716
2.6736	1	0.0287
2.0572	1	0.1582
2.6596	1	0.0299
3.6584	1	0.0015
2.4223	1	0.0591

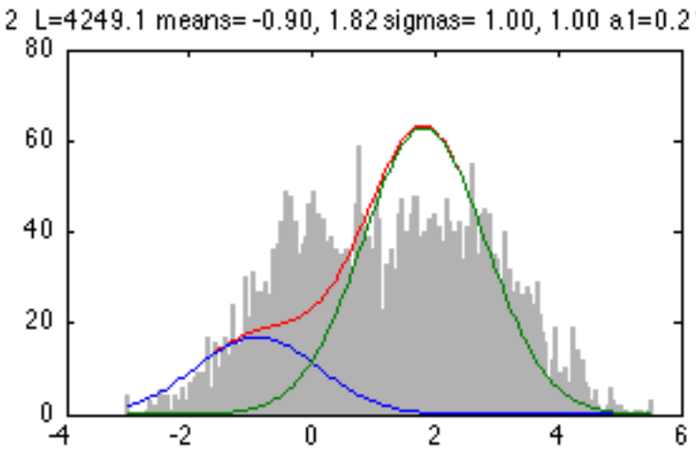


Example 2: Mixture of Gaussians

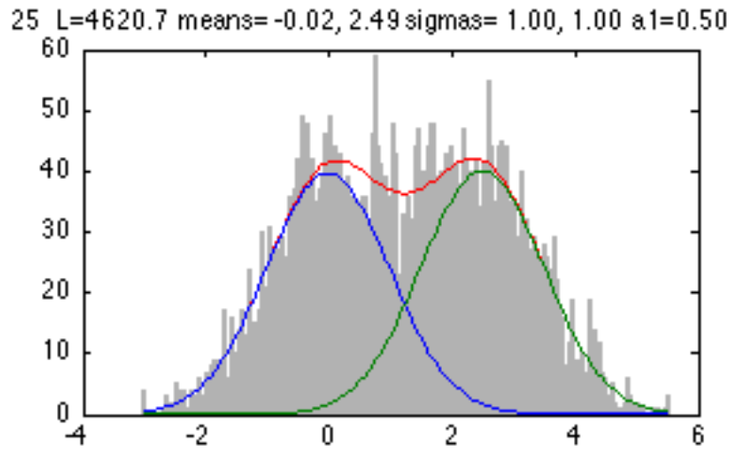
start

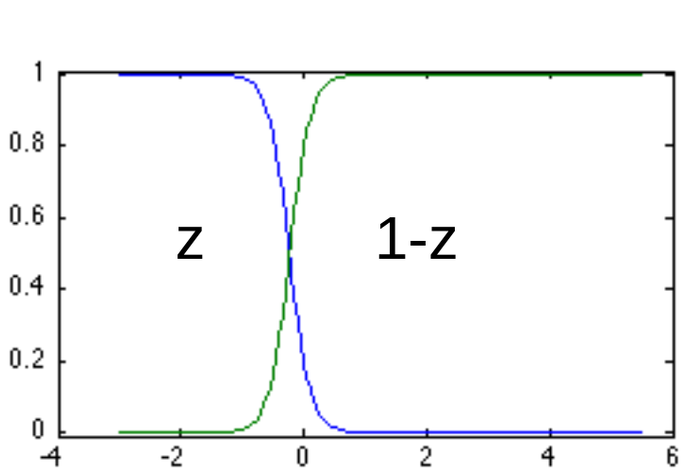


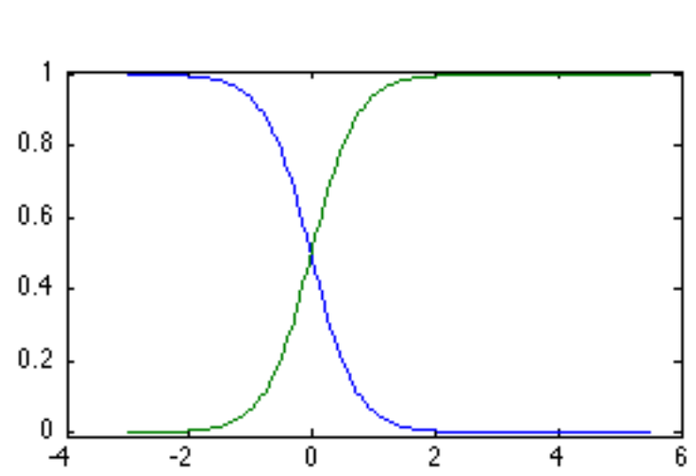
1 iteration

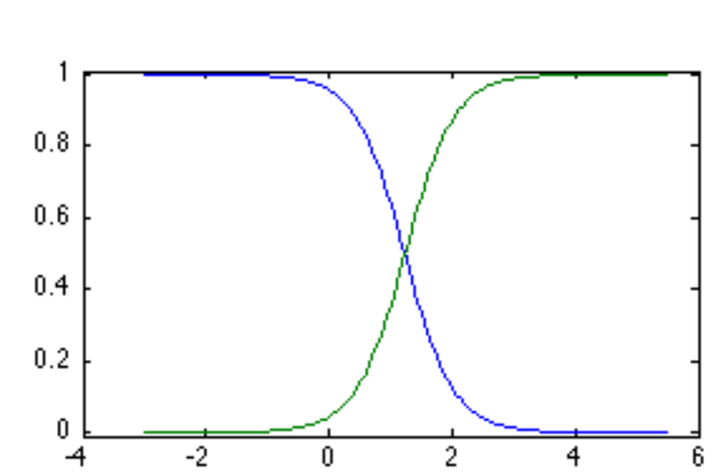


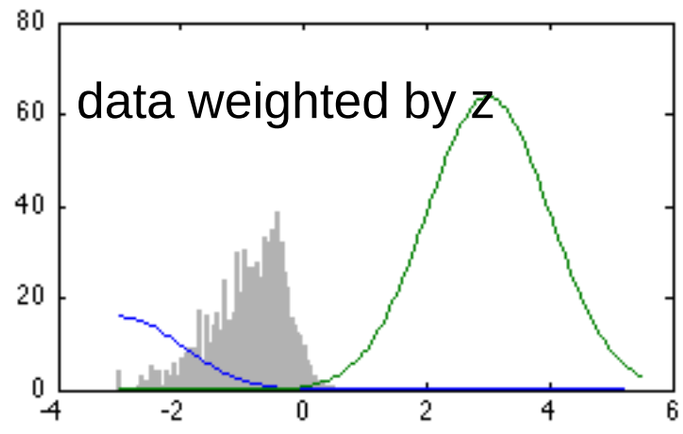
25 iterations

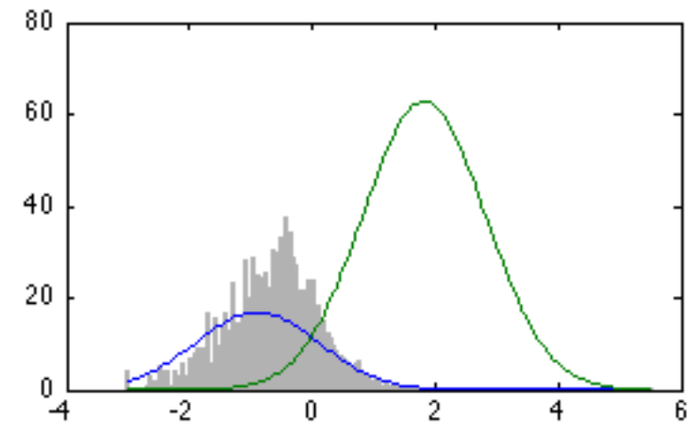


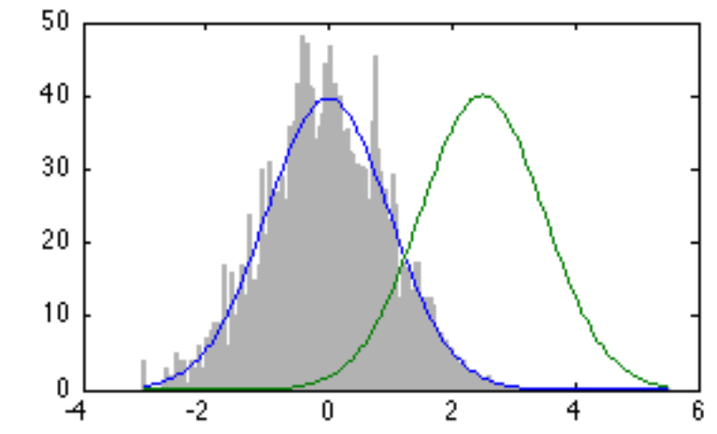












Likelihood with hidden variables

The general problem is formulated this way.

$\mathbf{x} = \{x_1, x_2 ... x_N\}$ is the set of observed data;

$\mathbf{z} = \{z_1, z_2, ... z_M\}$ is a set of hidden variables.

Suppose it's easy to compute the probability $p(\mathbf{x}, \mathbf{z} \mid \Theta)$ of the complete data set $\{x_1, x_2 ... x_N, z_1, z_2 ... z_M\}$.

By doing an integral over all values of the *z*'s we can get the log likelihood:

$$L = \ln \int p(\mathbf{x}, \mathbf{z} \mid \Theta)\, p(\mathbf{z} \mid \mathbf{x}, \Theta)\, d\mathbf{z}$$

The reconstruction problem

$\mathbf{x} = \{x_1, x_2 ... x_N\}$ is the set of images;

$\mathbf{z} = \{z_1, z_2, ... z_M\}$ is the set of alignment parameters (Euler angles and translations) for each image.

It is easy to compute the probability $p(\mathbf{x}, \mathbf{z} \mid \Theta)$ of the images, with the alignment parameters known.

In the end we don't care about the alignment parameters, and just integrate them out:

$$L = \ln \int p(\mathbf{x}, \mathbf{z} \mid \Theta)\, p(\mathbf{z} \mid \mathbf{x}, \Theta)\, d\mathbf{z}$$

But, maximizing *L* is still not simple...

The EM Algorithm

One way to increase the likelihood is to use the Expectation Maximization algorithm, which has two conceptual steps.

1. Given a previous estimate of the model parameters $\Theta^{(\text{old})}$, compute the expectation

$$Q(\Theta) = \int \ln p(\mathbf{x},\mathbf{z} \mid \Theta) p(\mathbf{z} \mid \mathbf{x},\Theta^{(\text{old})}) d\mathbf{z}$$

2. Maximize each parameter of the model by solving

$$\frac{\partial Q(\Theta^{(\text{new})})}{\partial L} = 0$$

Dempster Blair and Rubin, 1977

Example 3: 1D alignment

Suppose we have many instances x_i of a 1D signal buried in noise, and its relative time of arrival is z_i. Can we reconstruct it?

In this case Θ=y is the reconstructed signal and the Q function is

$$Q(y) = \sum_i \sum_z \left\| x_i - T_z y \right\|^2 p(z \mid x_i, y^{(old)})$$

and the EM iteration is

$$y^{(new)} = \sum_i \sum_z T_z x_i p(z \mid x_i, y^{(old)})$$

with the "switch variable" probability

$$p(z \mid x, y) = \frac{\exp\left(\frac{\left\| T_z x - y \right\|}{2\sigma^2} \right)}{\sum_z \exp\left(\frac{\left\| T_z x - y \right\|}{2\sigma^2} \right)}$$

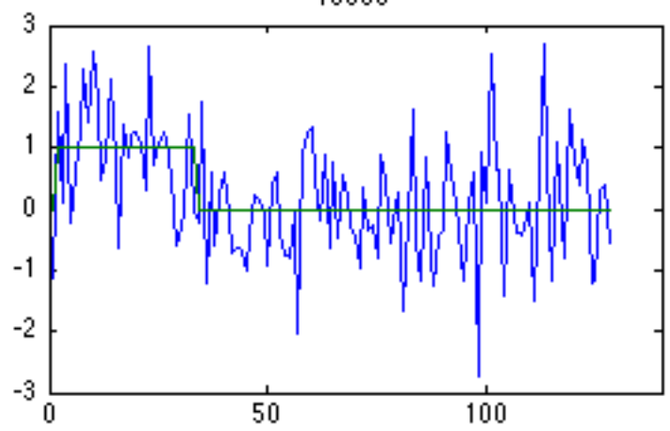
Example 3: 1D alignment

S/N=1

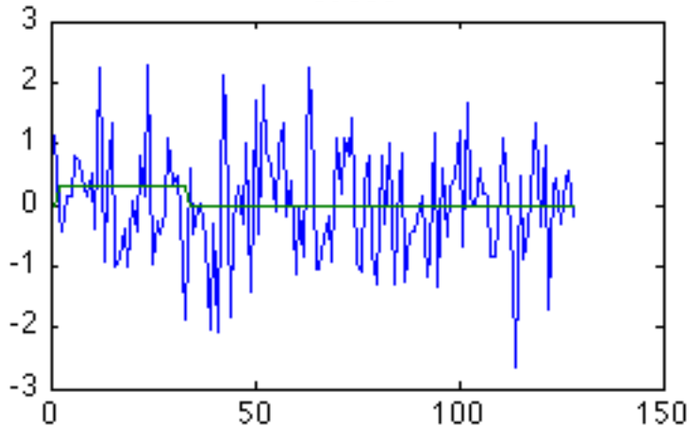
S/N=0.1

S/N=.01

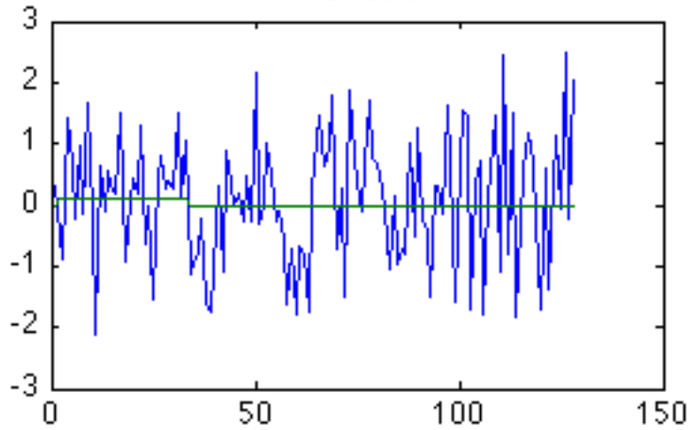
x_{10000}

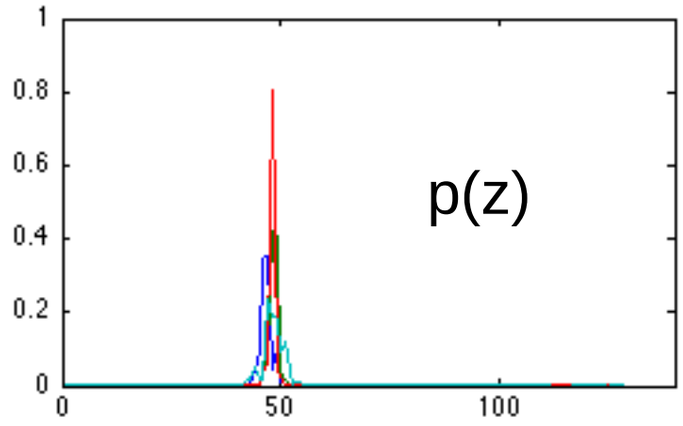


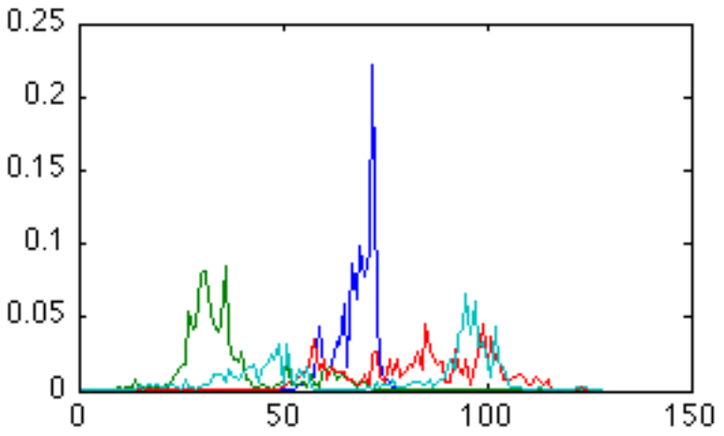
x_{85000}

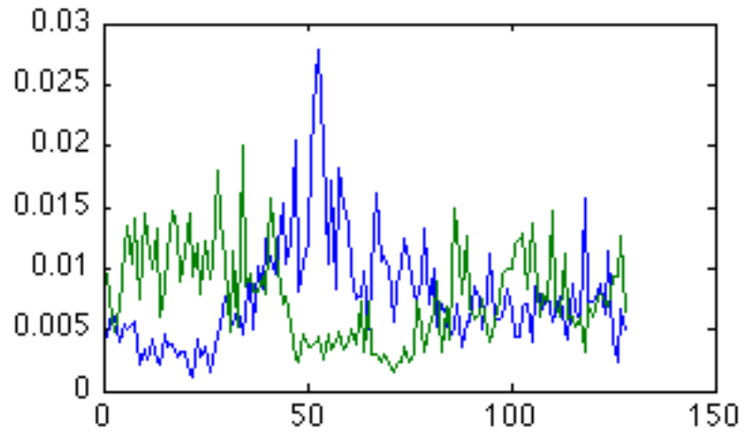


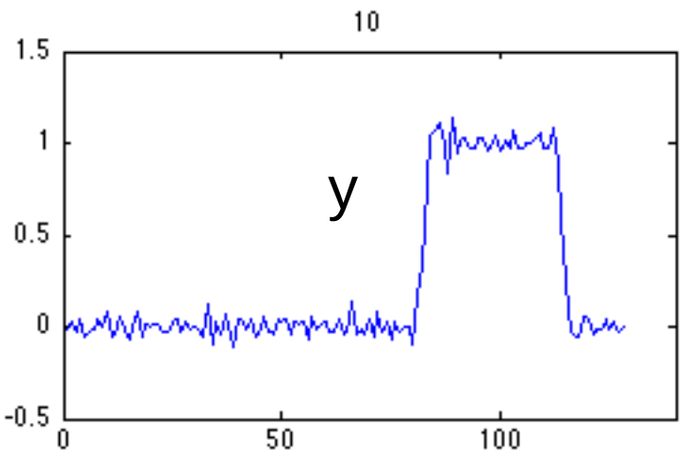
x_{300000}

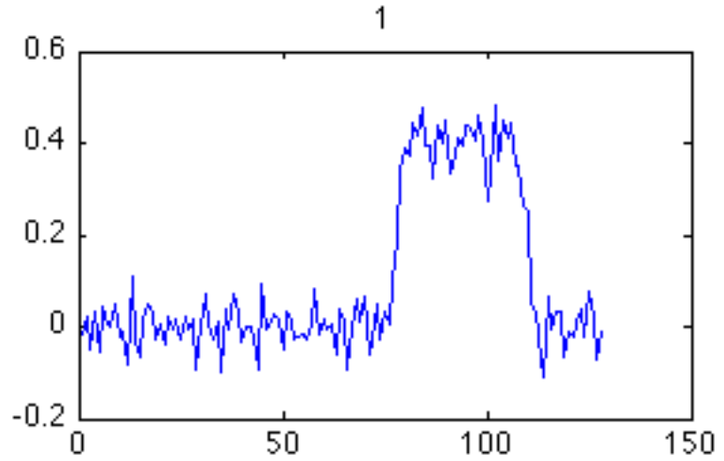


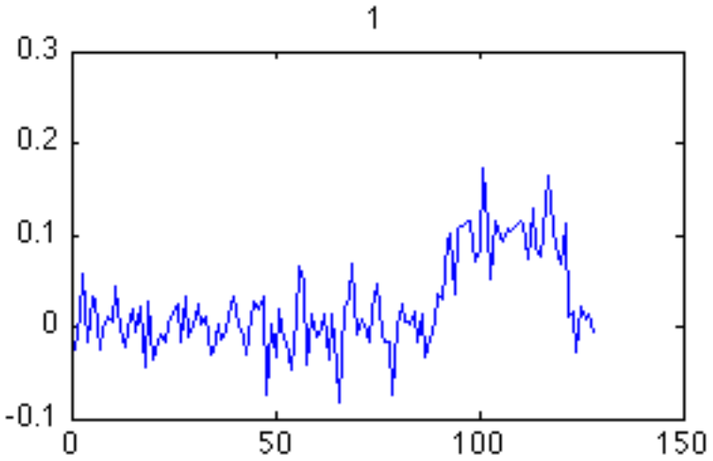












ML 3D Reconstruction

ML 3D reconstruction reduces to this problem:

Maximize the quantity (I've left out some constants)

$$Q = \sum_{i=1}^{N} \frac{\sum_{\kappa} \int \left\| x_i - R_{\varphi_i}(V_\kappa) \right\|^2 p(x_i, \phi, \kappa \mid \Theta^{(old)}) d\phi}{\sum_{\kappa} \int p(x_i, \phi, \kappa \mid \Theta^{(old)}) d\phi}$$

Q is maximized with respect to each voxel of each reconstructed volume, plus a few other parameters (e.g. σ). The maximization was done using an algebraic reconstruction technique by Scheres, Carazo et al.

Notice that both the numerator and denominator involve an integral over all five alignment parameters and a sum over the conformations.

$$\Theta = \left\{ V_1, V_2, \ldots, a_1, a_2, \ldots, \sigma \right\}$$

$p(\mathbf{x},\mathbf{z}\mid\Theta)$ is easy to compute??

Assuming independent Gaussian noise in each of *P* pixels in an image, the probability for one image is

$$p(x_i,\phi_i,\kappa_i \mid \Theta) = \left(\frac{\varepsilon}{\sqrt{2\pi\sigma}}\right)^P \exp\left(-\frac{\left\|x_i - R_{\varphi_i}(V_\kappa)\right\|^2}{2\sigma^2}\right)$$

where

x_i is the image

φ_i,κ_i are the corresponding hidden parameters (alignment, conformation)

R_φ is the projection operator

V_κ is the k^{th} reconstruction volume (an element of Θ)

The other quantity we need for the EM algorithm is the probability of the hidden variables

$$p(\phi,\kappa \mid x_i,\Theta) = \frac{p(x_i,\phi,\kappa \mid \Theta)}{\sum_\kappa \int p(x_i,\phi,\kappa \mid \Theta)d\phi}$$

ML 3D Reconstruction

ML 3D reconstruction reduces to this problem:

Maximize the quantity (I've left out some constants)

$$Q = \sum_{i=1}^{N} \frac{\sum_{\kappa} \int \left\| x_i - R_{\varphi_i}(V_\kappa) \right\|^2 p(x_i, \phi, \kappa \mid \Theta^{(old)}) d\phi}{\sum_{\kappa} \int p(x_i, \phi, \kappa \mid \Theta^{(old)}) d\phi}$$

Q is maximized with respect to each voxel of each reconstructed volume, plus a few other parameters (e.g. σ). The maximization was done using an algebraic reconstruction technique by Scheres, Carazo et al.

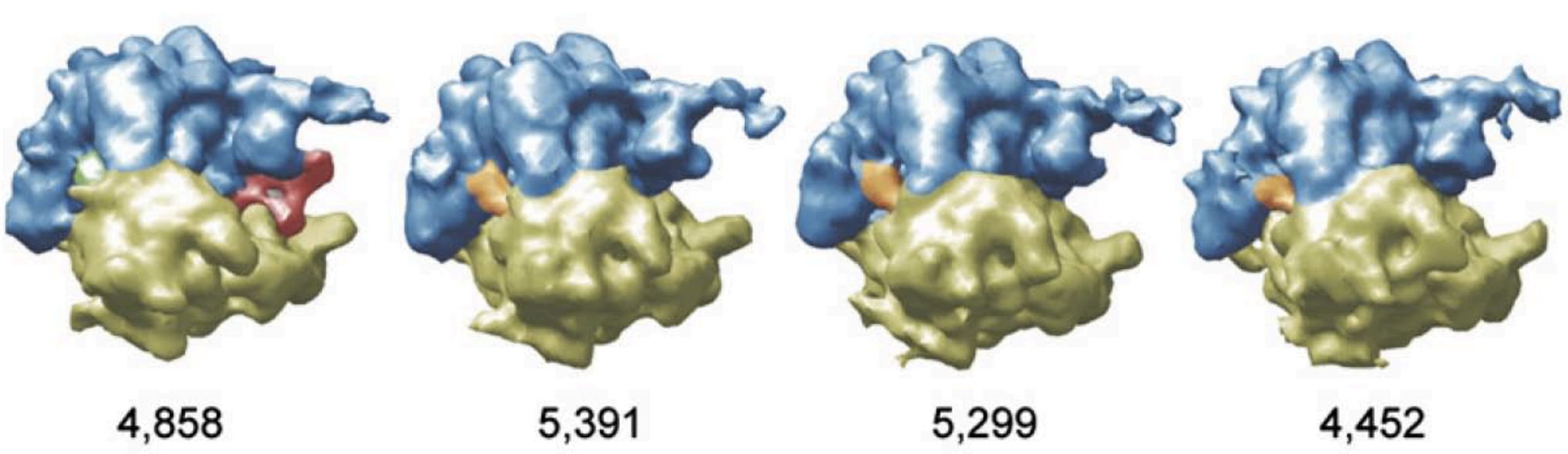
Notice that both the numerator and denominator involve an integral over all five alignment parameters and a sum over the conformations.

ML 3D Reconstruction

Structure
Ways & Means

Modeling Experimental Image Formation for Likelihood-Based Classification of Electron Microscopy Data

Sjors H.W. Scheres,[1] Rafael Núñez-Ramírez,[1] Yacob Gómez-Llorente,[1] Carmen San Martín,[1] Paul P.B. Eggermont,[2] and José María Carazo[1,*]



MLE and MAP Estimation

The probability of the model is related to the likelihood by Bayes' theorem,

$$p(\Theta \mid \mathbf{x}) = p(\mathbf{x} \mid \Theta)\frac{p(\Theta)}{p(\mathbf{x})}$$

The maximum-likelihood estimate (MLE) optimizes $p(\mathbf{x} \mid \Theta)$.

$$\text{Experiment} \longrightarrow \Theta$$

The maximum a posteriori estimate (MAP) optimizes $p(\mathbf{x} \mid \Theta)p(\Theta)$.

$$p(\Theta) \longrightarrow \text{Experiment} \longrightarrow \Theta$$

a priori *a posteriori*

MAP Estimation

For example, suppose we know that our particle images are approximately centered. We could include this prior knowledge as a prior probability term

$$p(t_x, t_y) = \frac{1}{\sqrt{2\pi\sigma_t}} \exp\left(-\frac{t_x^2 + t_y^2}{2\sigma_t^2} \right)$$

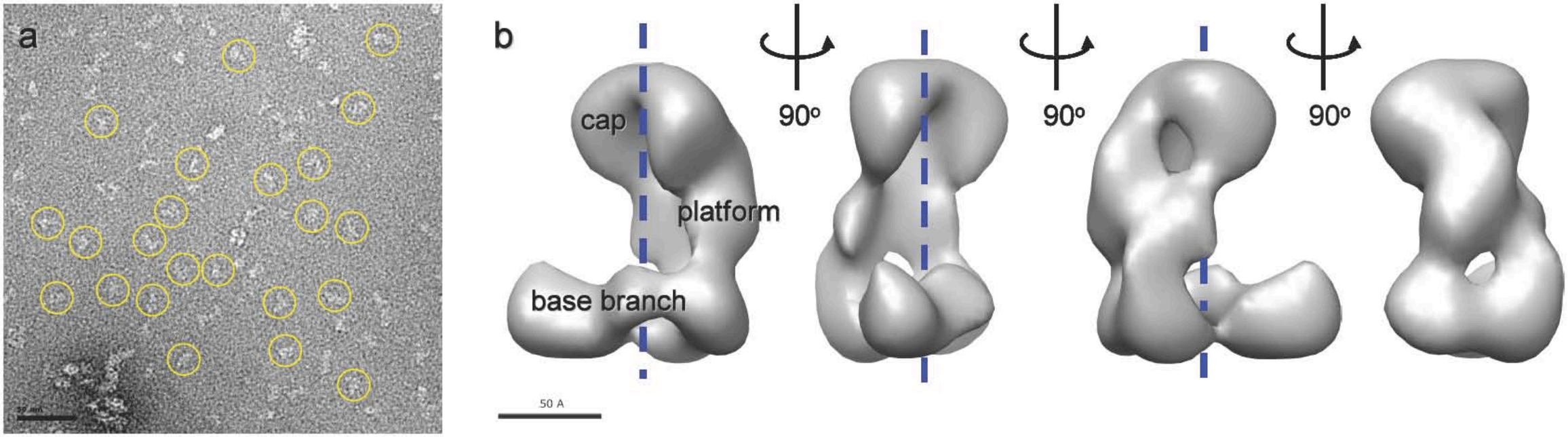
Another example would be the prior knowledge that outside our particle volume the density is constant (solvent flattening). Formally, we would say that the prior probability is low whenever the voxels outside the volume are nonzero. In the EM algorithm this is imposed simply by zeroing these voxels at each iteration of reconstruction.

2. Flexible particle reconstruction

Fred Sigworth
Hemant Tagare
Hongwei Wang

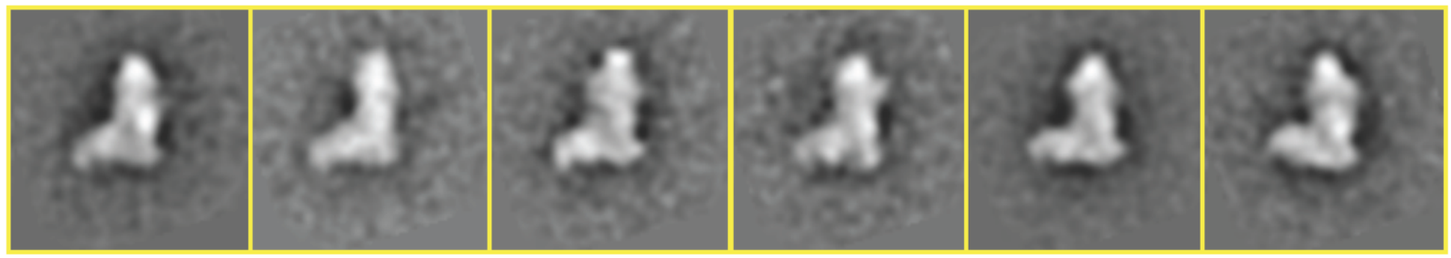
Yale University

Dicer is a 200 kDa RNA-processing protein

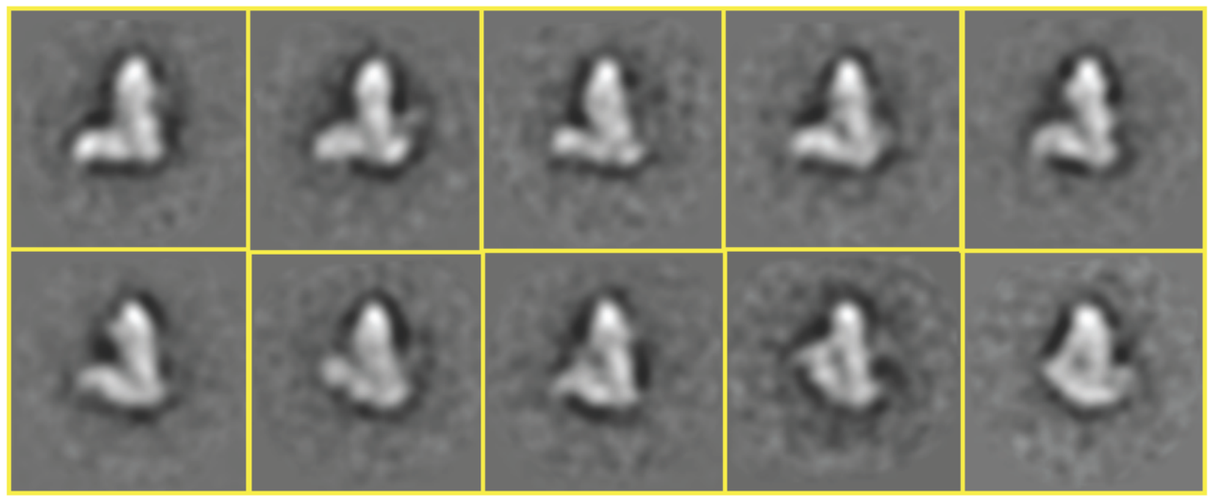


Architecture of human Dicer. (a) Negative-stain micrograph. (b) 3D reconstruction of human Dicer shown in four different orientations.

Depending on the substrate, Dicer shows flexibility



Class averages of a preferred view of Dicer alone,
in negative stain.

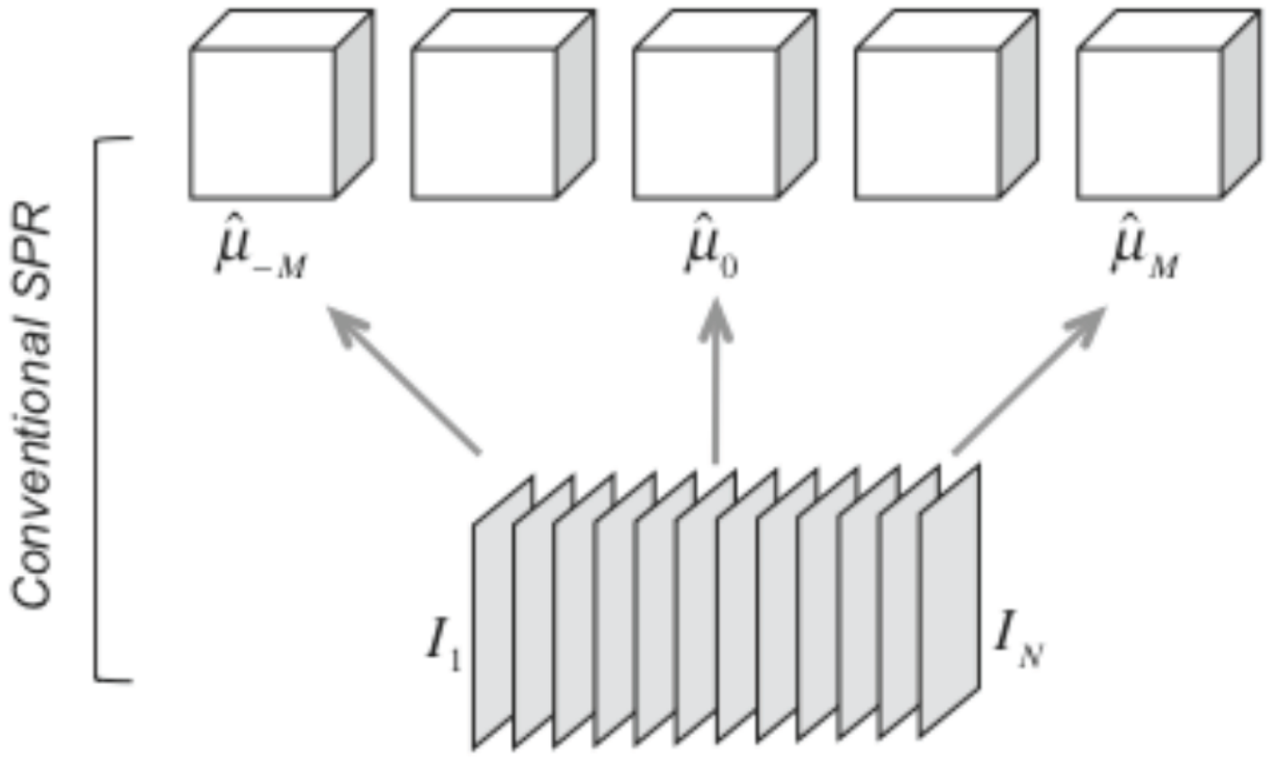


Class averages of Dicer + RNA substrate

In the conventional method, multiple 3D reconstructions are made from a set of data images.

From the *N* images, a set of *M* reconstructions are made.

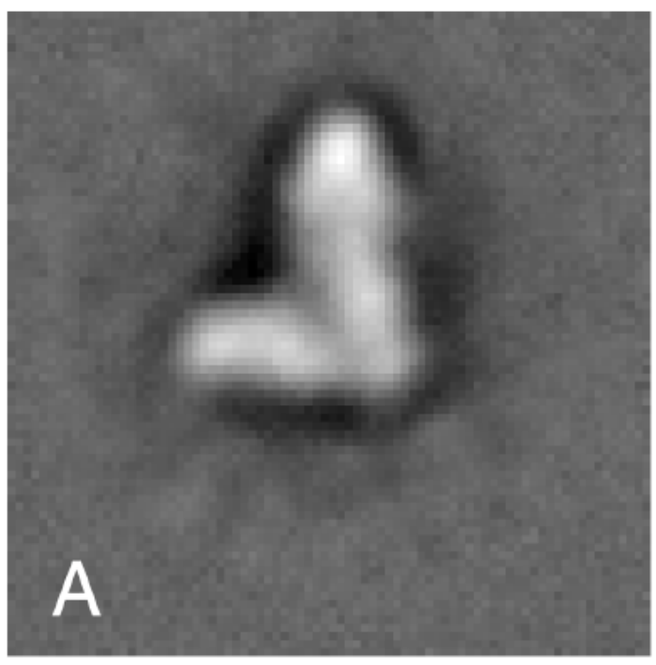
But there is a penalty for having *M* large, as only ~*N* / *M* images contribute to each reconstruction.

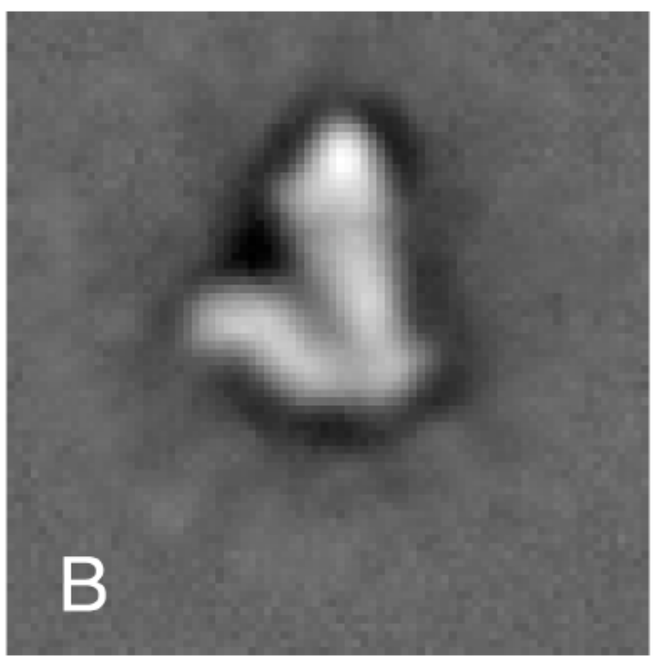


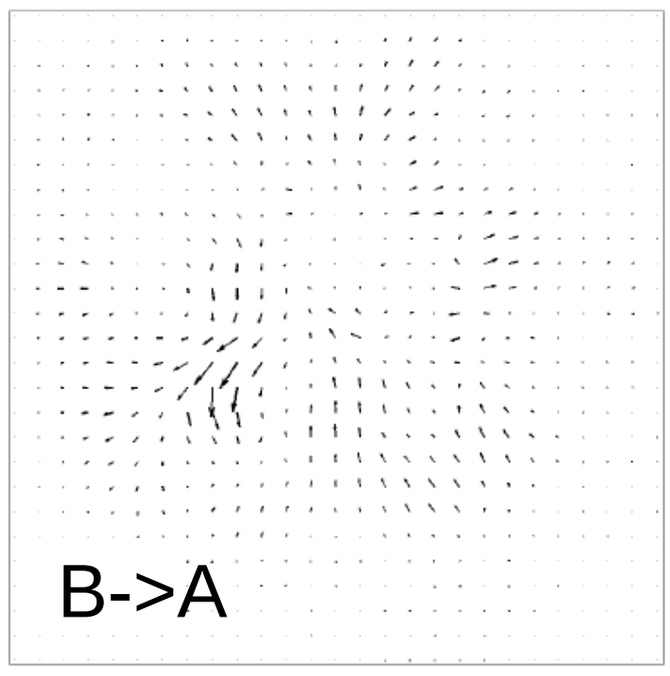
Proposal: use a continuous “warping” to describe motions

A diffeomorphism (differentiable and invertible warping) is used to map each of a set of structures to a reference state. The reference state could then be reconstructed to high resolution.

Here is an example of a two-dimensional diffeomorphism:





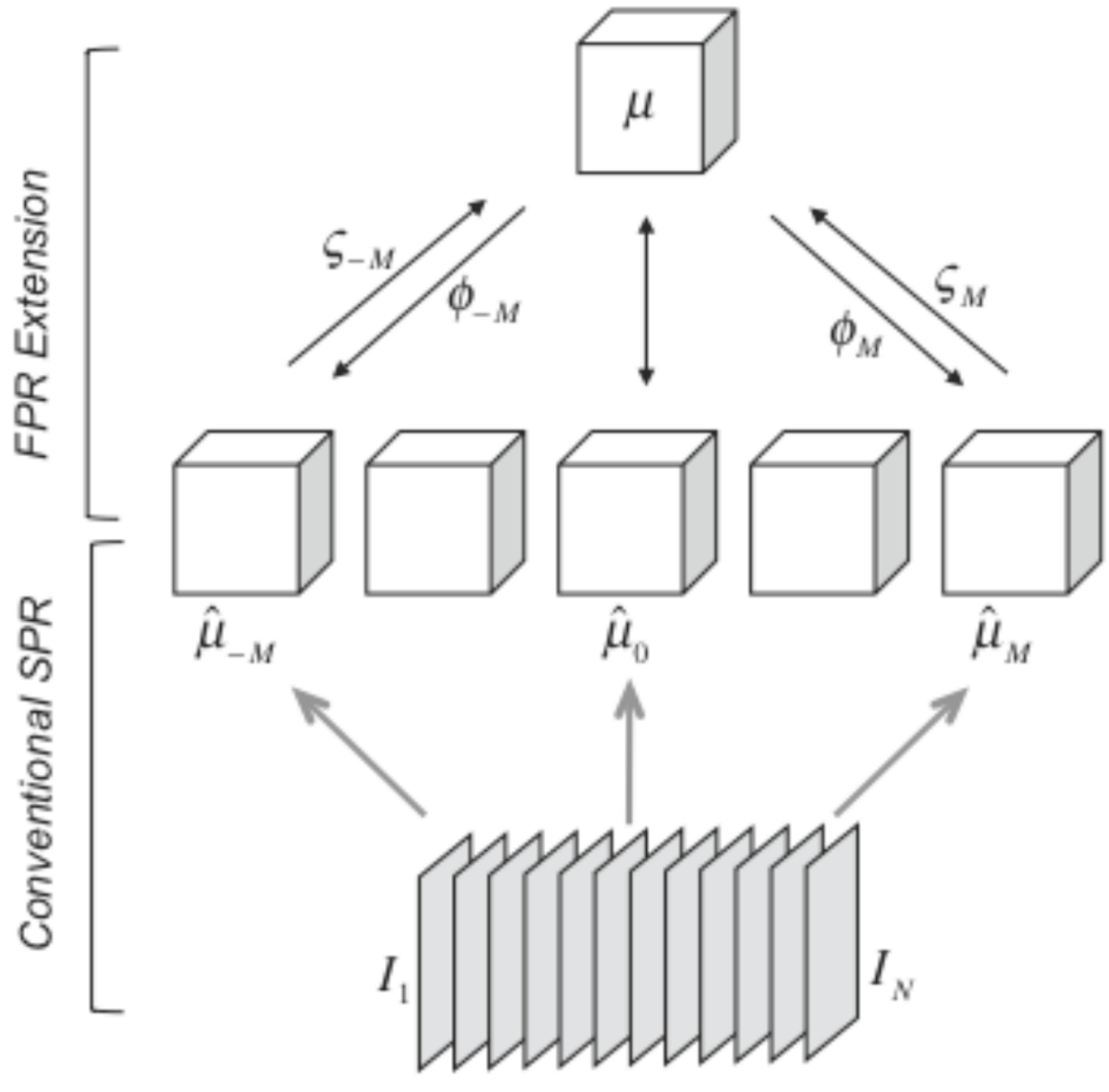


The 3D reconstruction algorithm

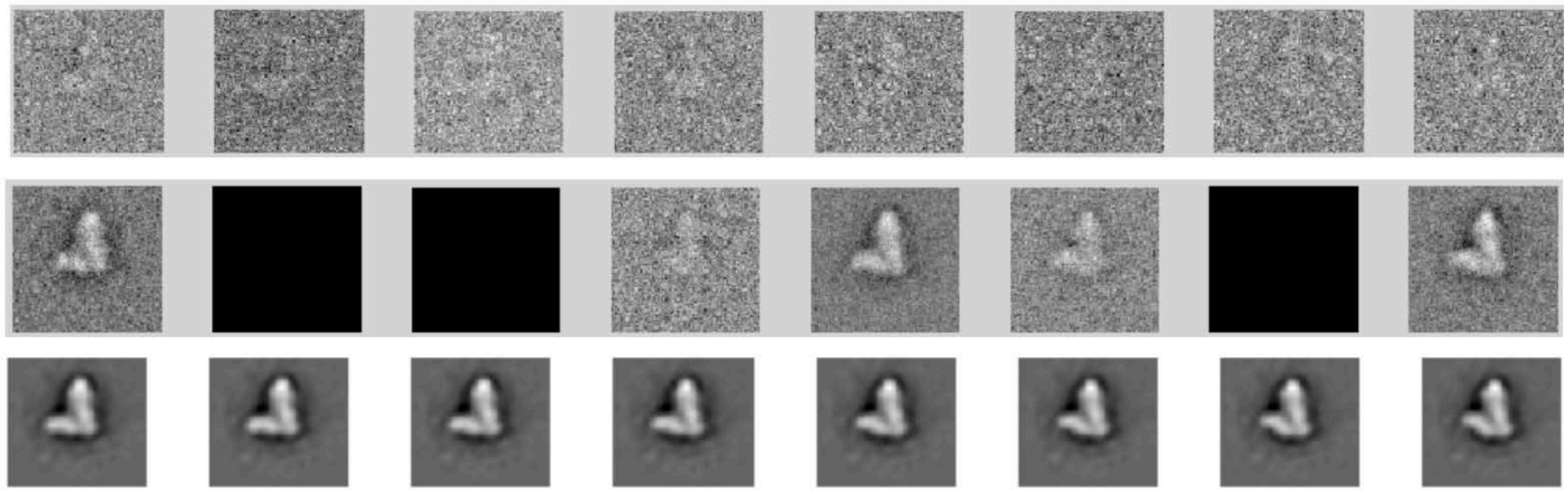
During reconstruction, the μ_k are constrained to all be warpings of the “flex mean” μ.

Iterations optimize an objective function that enforces the warping constraints, and also ensures that the warpings are smooth.

This 3D algorithm hasn’t been implemented yet.



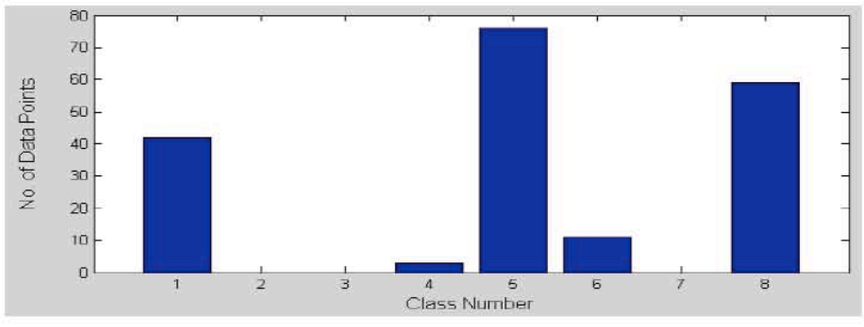
Example of the algorithm in 2D, on Dicer preferred views.

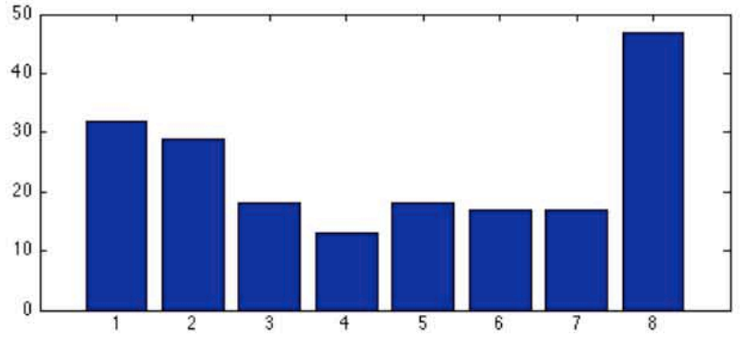


Raw images

Initial class means μ_k

Final class means

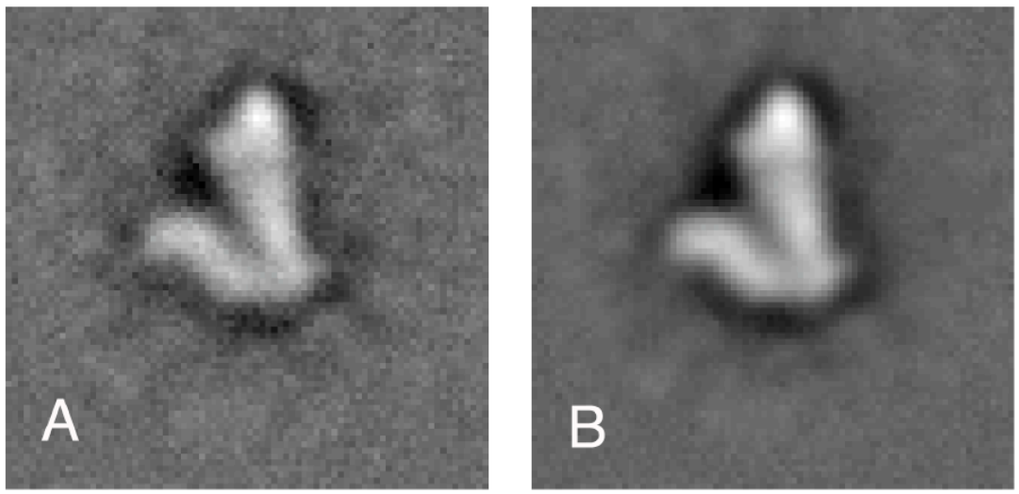




Class membership, initially and finally.

For the first iteration, the μ_k were initialized by adding a variable amount of the top PCA eigenimage to the global mean.

The flex mean reconstruction is better.



Improved reconstruction quality with FPR. A, a single class mean from standard SPR. B, the FPR result, the warped mean $\phi \circ \mu$.

See the poster by Hemant Tagare et al. for more details.