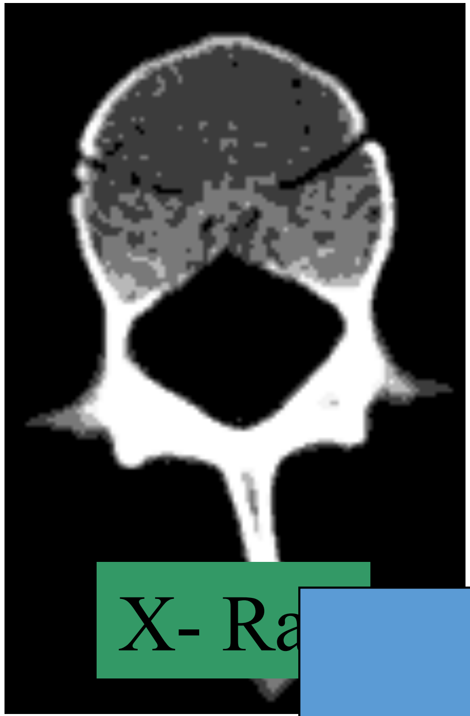
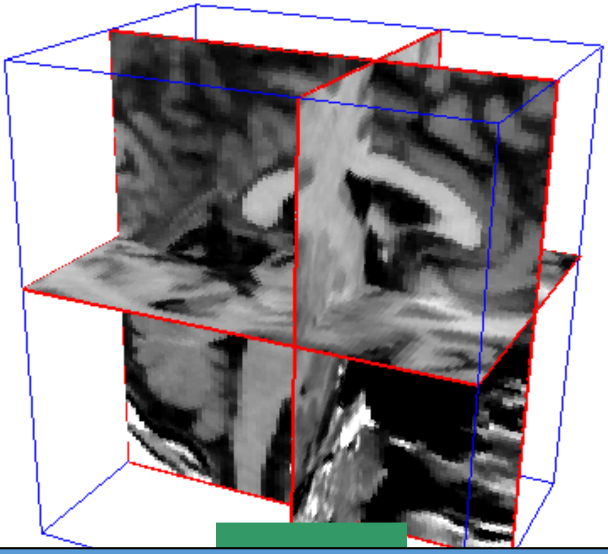
Medical Imaging :
Image Segmentation & Classification

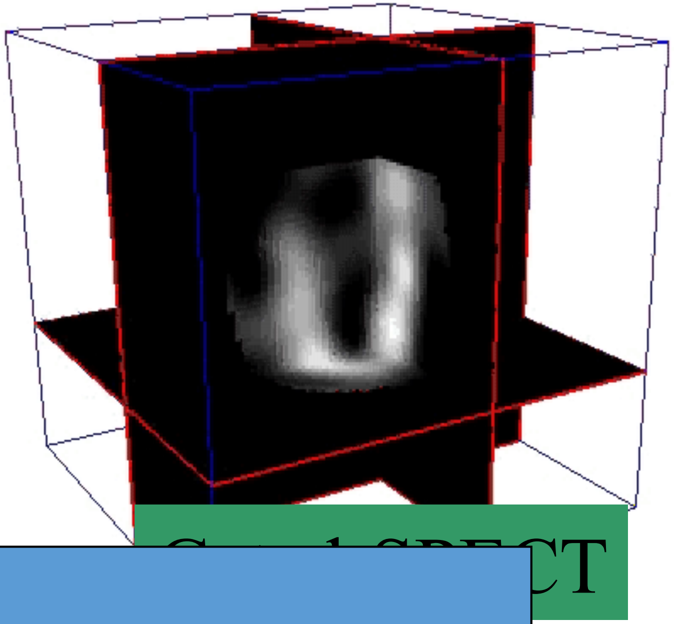
Hervé Delingette
Epione Team
Herve.Delingette@inria.fr

3. Medical Image Segmentation

Image Segmentation

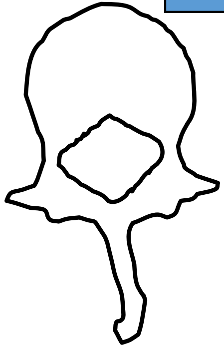


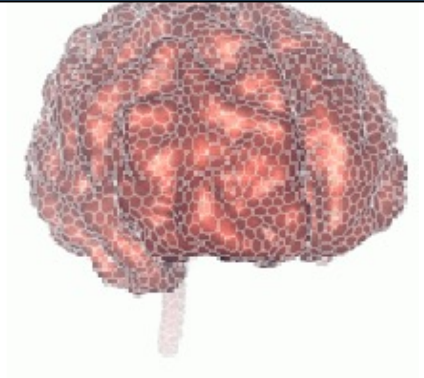


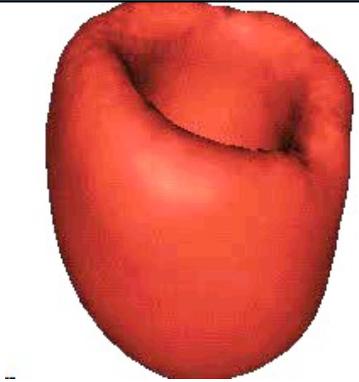


Isolate a Region of Interest in a Medical Image

2D

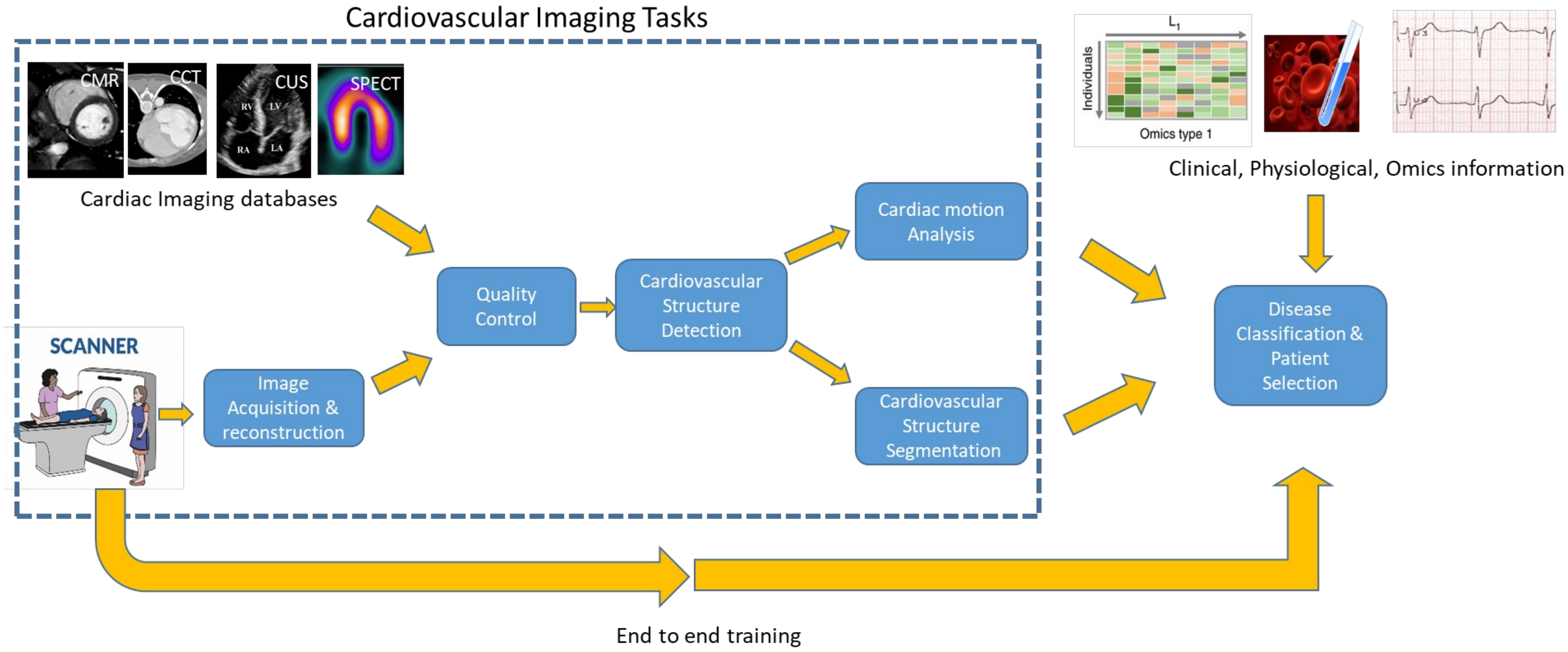






Segmentation in clinical workflow

- Example of cardiovascular Imaging



Cardiovascular Imaging Tasks

Cardiac Imaging databases

Image Acquisition & reconstruction

Quality Control

Cardiovascular Structure Detection

Cardiac motion Analysis

Cardiovascular Structure Segmentation

Clinical, Physiological, Omics information

Disease Classification & Patient Selection

End to end training

Segmentation Algorithms

- Various taxonomy of segmentation algorithms :
  - Discrete vs Continuous
  - Bottom-up vs Top-down approaches
  - Boundary vs Region approaches
  - Supervised or non supervised
  - Intensity or Shape based

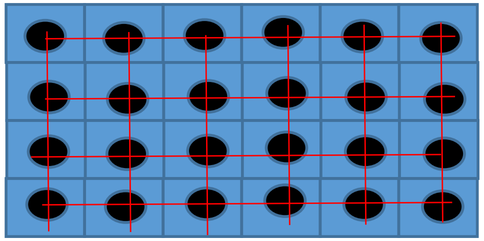
Discrete vs Continuous Image Representation

$I(x)$ **Image Domain or Image Value** can be either discrete or continuous

Image Domain Image Value	**Discrete**	**Continuous**
Discrete	Array of Int	~~Field of Integer~~
Continuous	Array of Float	Field of Float

Discrete Image Representation

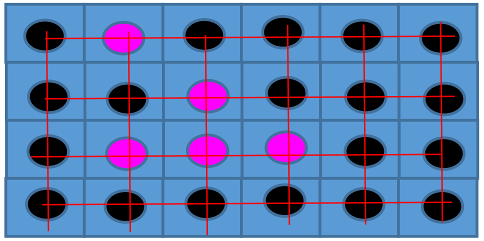
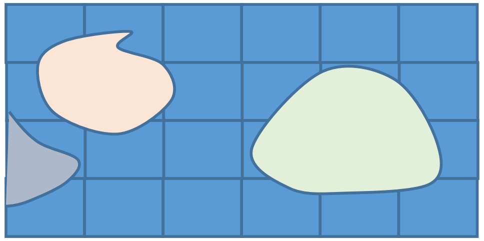
- Image as a 2D or 3D array
- Representation I[row][col]
- Image can be seen as a graph



Continuous Image Representation

- Image as a 2D or 3D field I(x)
- Requires definition of Interpolation and Extrapolation functions :
    - Nearest Neighbor Interpolation
    - Bi(Tri)Linear Interpolation
    - (Cubic)Spline Interpolation

Discrete vs Continuous Image Segmentations

Segmentation with Discrete Image Representation	Segmentation with Continuous Image Representation
Define a binary variable $z_n \in \{0,1\}$ • $z_n = 1$ if pixel is in foreground • $z_n = 0$ if pixel is in background	Define Regions $\{\Omega_i\}$ inside which a structure is defined
Can be generalized to a set of Labels $\mathcal{L} = \{0,1,\ldots,M\}$	⟺ Define close or open contours $\{\partial\Omega_i\}$ Separating background from structure i
	
Segmentation obtained through discrete/ combinatorial optimization	Segmentation obtained through variational principles (calculus of variations…)

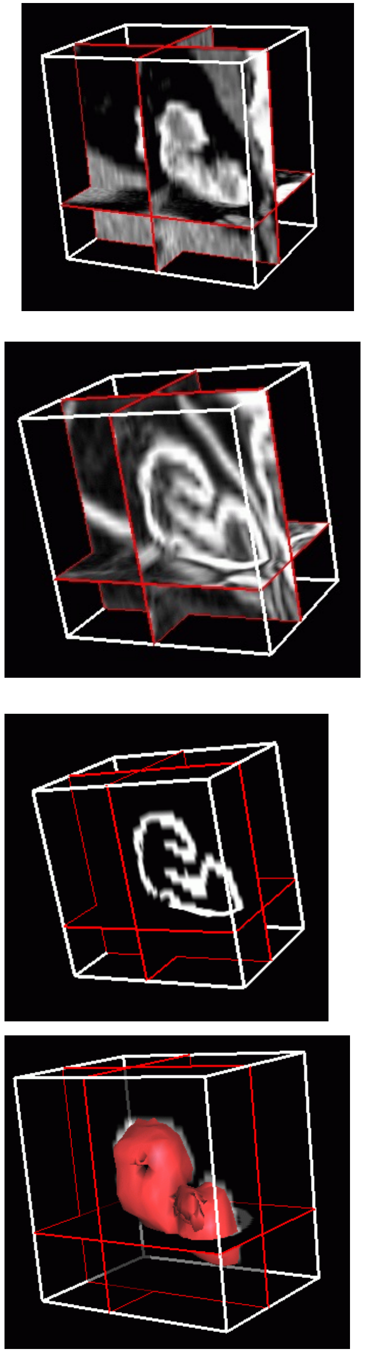
Bottom-up Approach

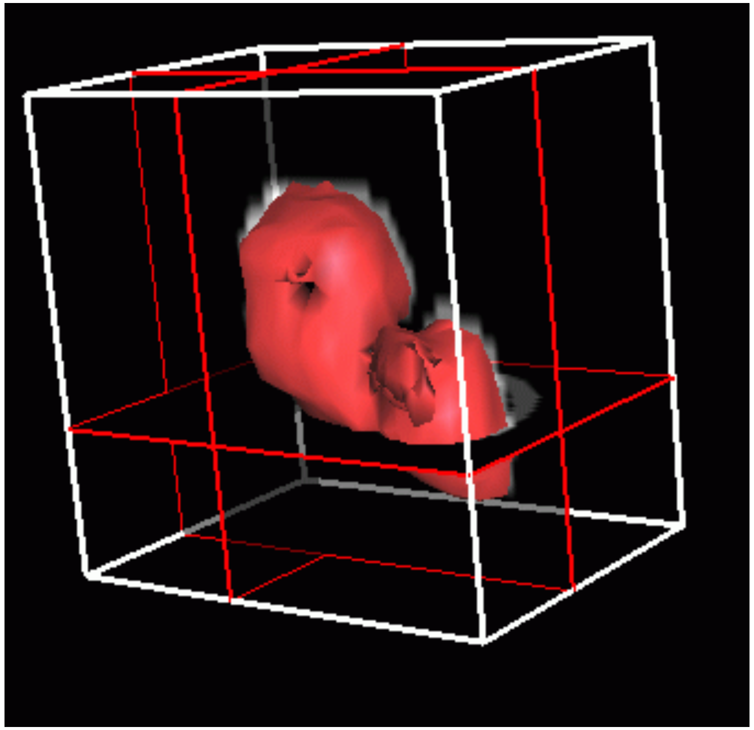
Medical Image

Feature Extraction

Feature Grouping

Region/Boundary Extraction



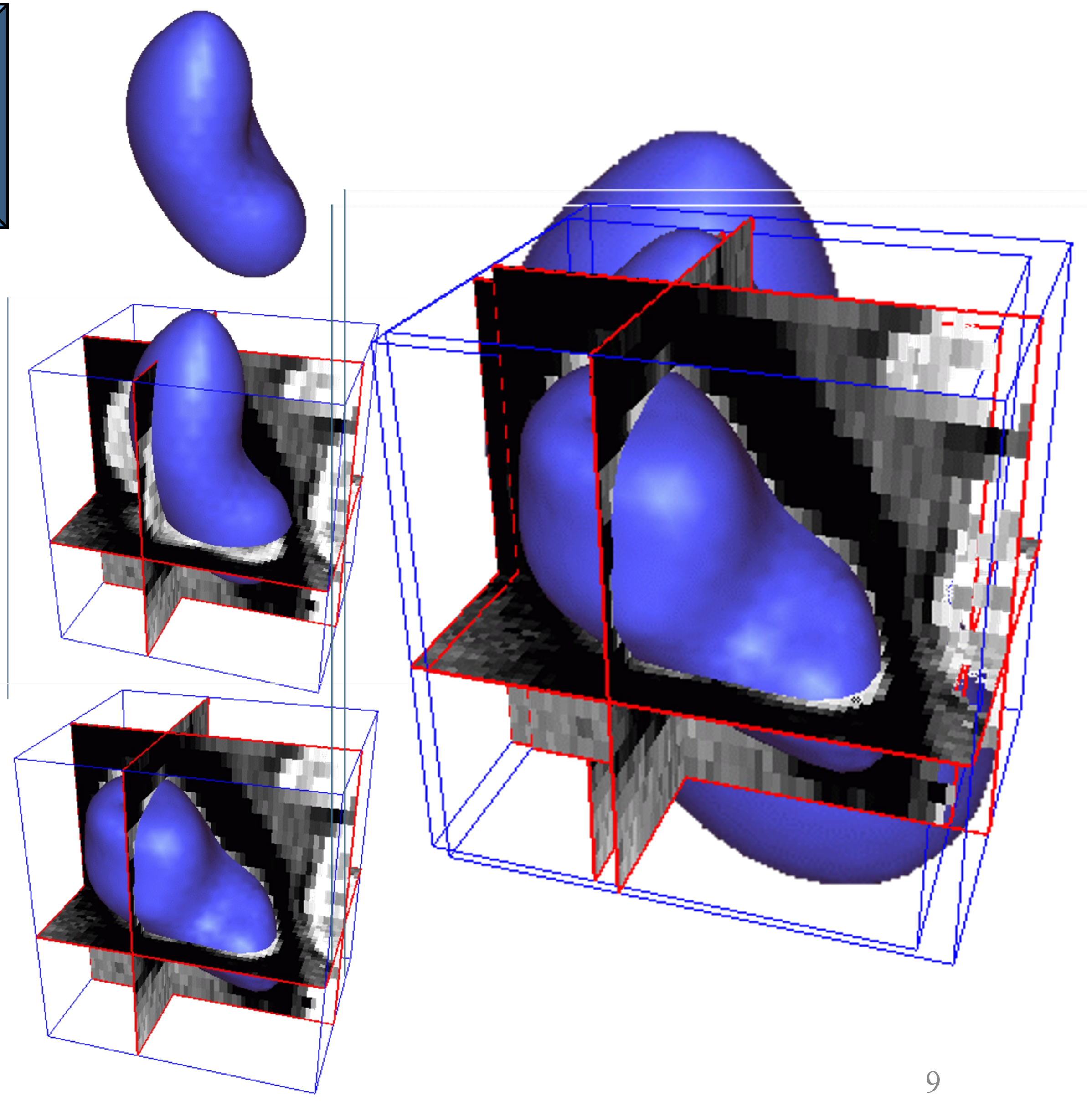


Top-down approach

Model Construction :
Shape and Appearance

Model Initialisation

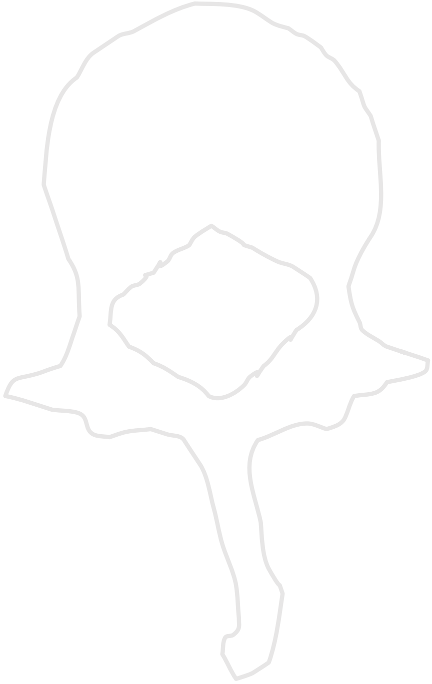
Model Optimization



Region vs Boundary Methods

Image

Region-based segmentation

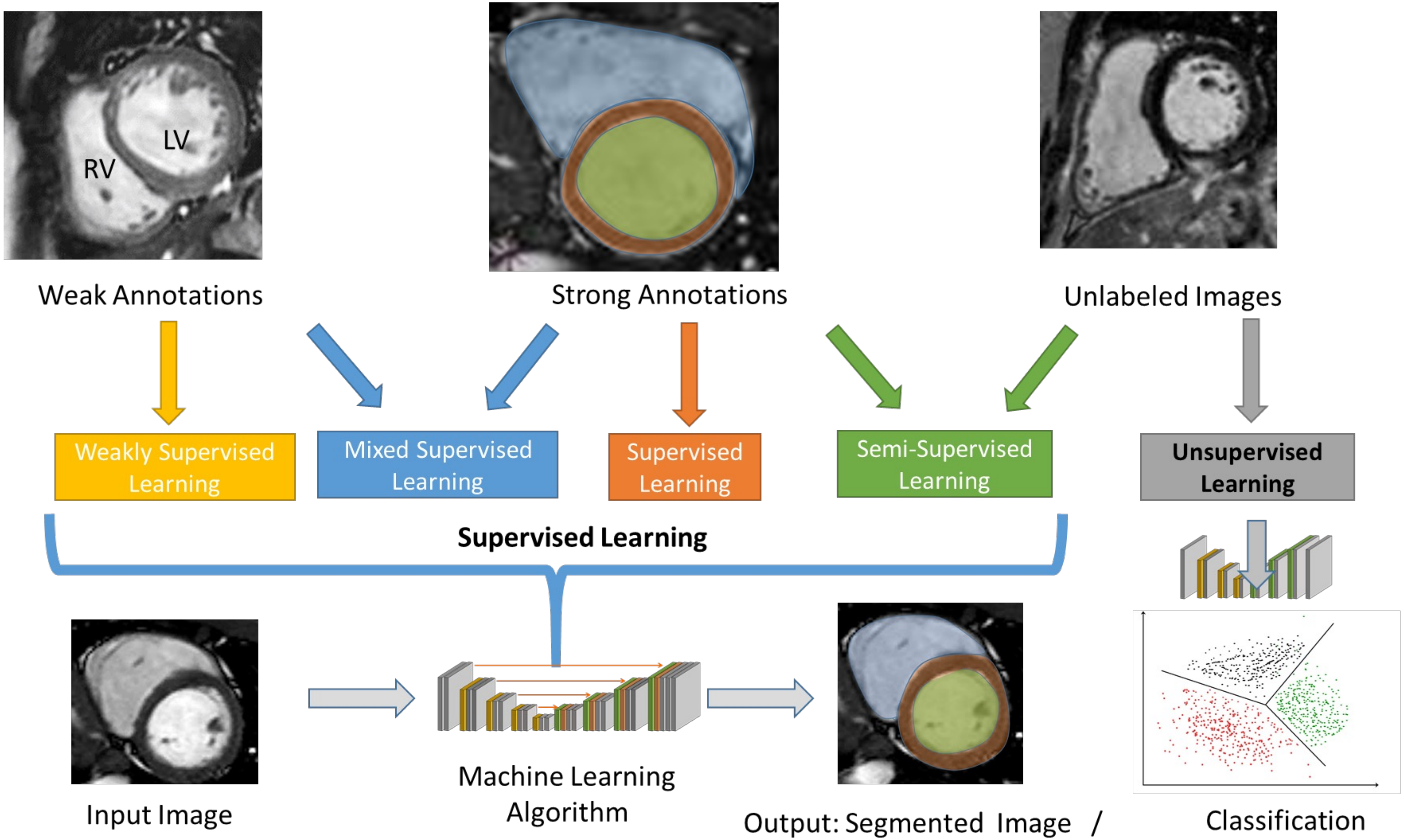


Boundary-based segmentation

Supervision of Image Segmentation

- Supervised Image Segmentation Problems:
  - Several examples of image segmentations are available
  - Methods : machine learning, multi-atlas registration
  - Very costly to produce annotated data
- Unsupervised Image Segmentation Problems :
  - No examples are available
  - Models of image content and shape are used to produce image segmentation
- Weakly supervised Segmentation Problems :
  - Only partial labels are available
- Semi supervised Segmentation Problems :
  - Fully annotated images and images with no annotations
- Mixed supervised Segmentation Problem :
  - Fully annotated images and weakly annotated images

Supervision of Image Segmentation



Weak Annotations

Strong Annotations

Unlabeled Images

Weakly Supervised Learning

Mixed Supervised Learning

Supervised Learning

Semi-Supervised Learning

Unsupervised Learning

Supervised Learning

Input Image

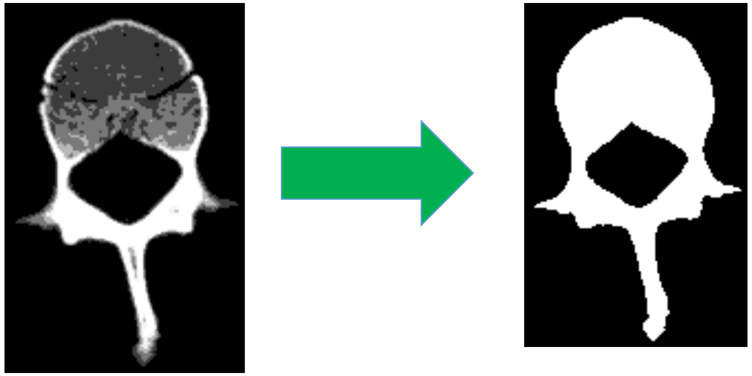
Machine Learning Algorithm

Output: Segmented Image / Classification

Image Segmentation Approaches

Intensity Only

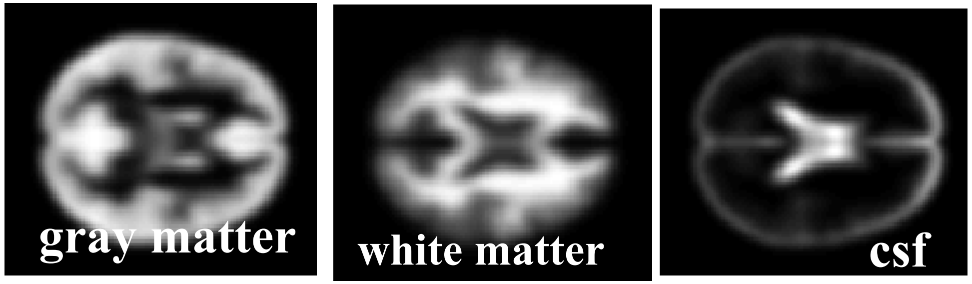
Thresholding



Contrast Agent in CT

Lesions in CT / MR

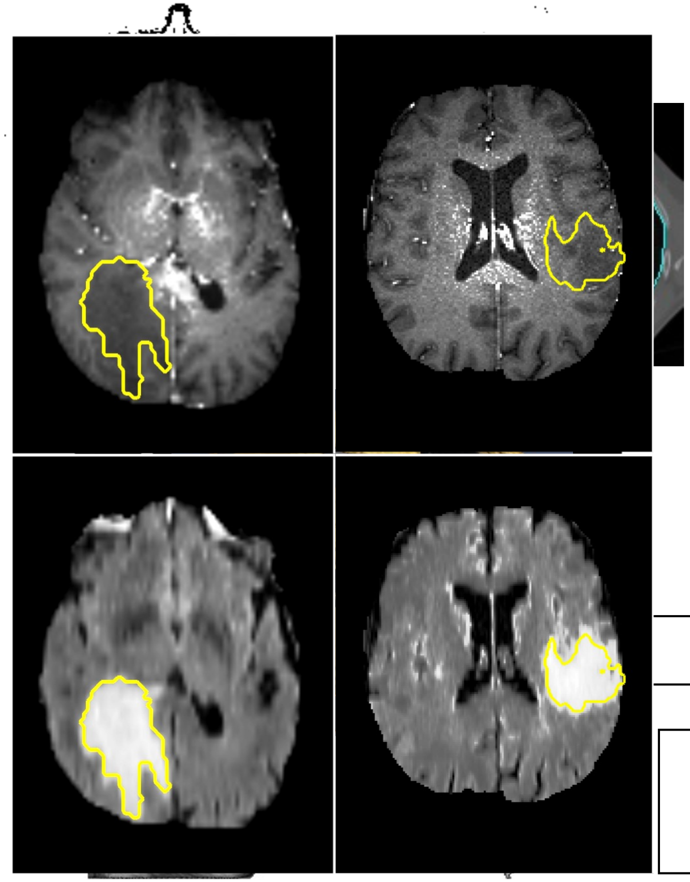
Intensity-only Classification



No Typical Shape

Typical Shape

Image Segmentation Approaches



Intensity and connexity between regions

Vessels / tumors / bones /lesions

Grey / White matter in MR

Mathematical Morphology

MRF (graph cuts, RW, watershed)

Machine Learning (Deep Learning, RF, SVM, ,ML)

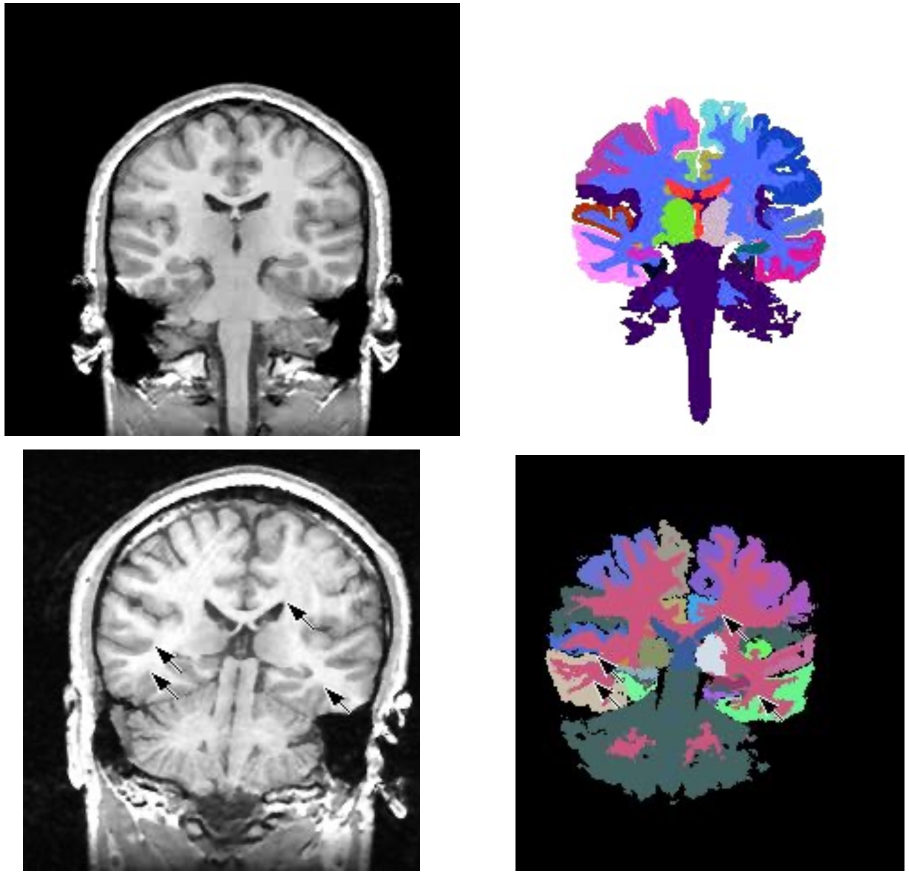
No Typical Shape

Typical Shape

Camille Couprie, Leo Grady, Laurent Najman and Hugues Talbot , **"Power watershed: A Unifying Graph-Based Optimization Framework"** , IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33 (7), pp 1384-1399 (2011)

Ezequiel Geremia, Olivier Clatz, Bjoern H. Menze, Ender Konukoglu, Antonio Criminisi, and Nicholas Ayache. **Spatial Decision Forests for MS Lesion Segmentation in Multi-Channel Magnetic Resonance Images**. *NeuroImage*, 57(2):378-90, July 2011

Image Segmentation Approaches



Atlas based Segmentation

Intensity and shape

Heart

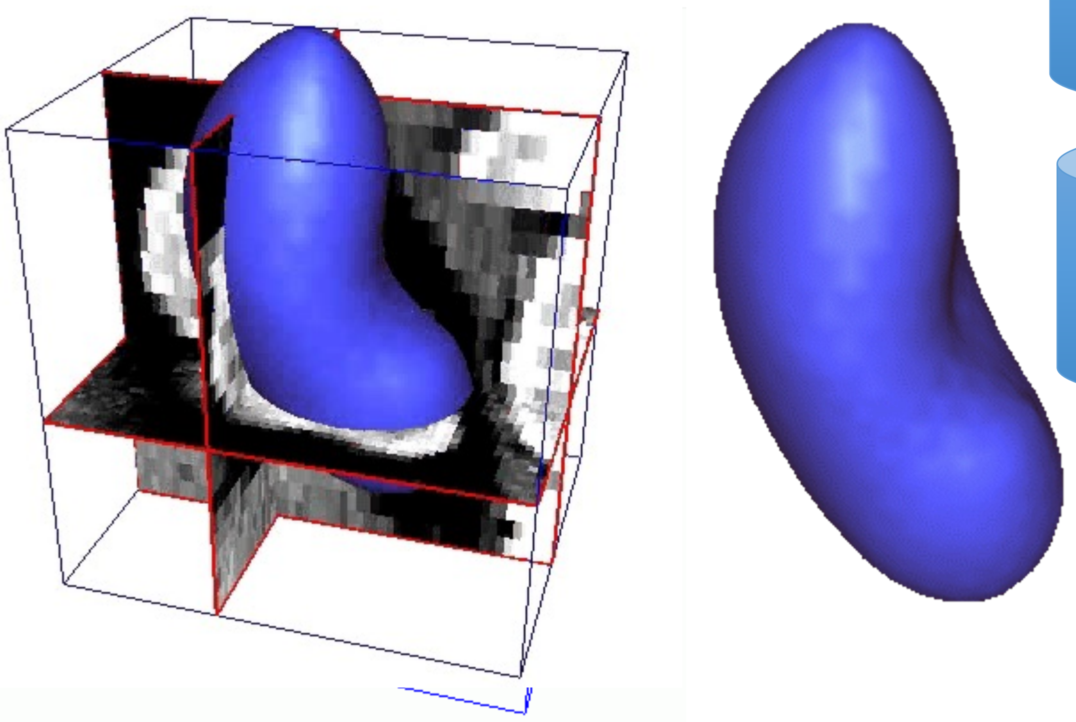
Liver

Bones

No Typical Shape

Typical Shape

Image Segmentation Approaches



Atlas based Segmentation

Deformable Templates

Intensity and shape

Heart

Liver

Bones

No Typical Shape

Typical Shape

3. Medical Image Segmentation

3.1 Taxonomy of segmentation algorithms

3.2 **Validation of segmentation algorithms**

3.3 Deterministic Filtering & Thresholding Approaches

3.4 Probabilistic Imaging Model

3.5 Expectation Maximisation for GMM

3.6 Image classification with bias field

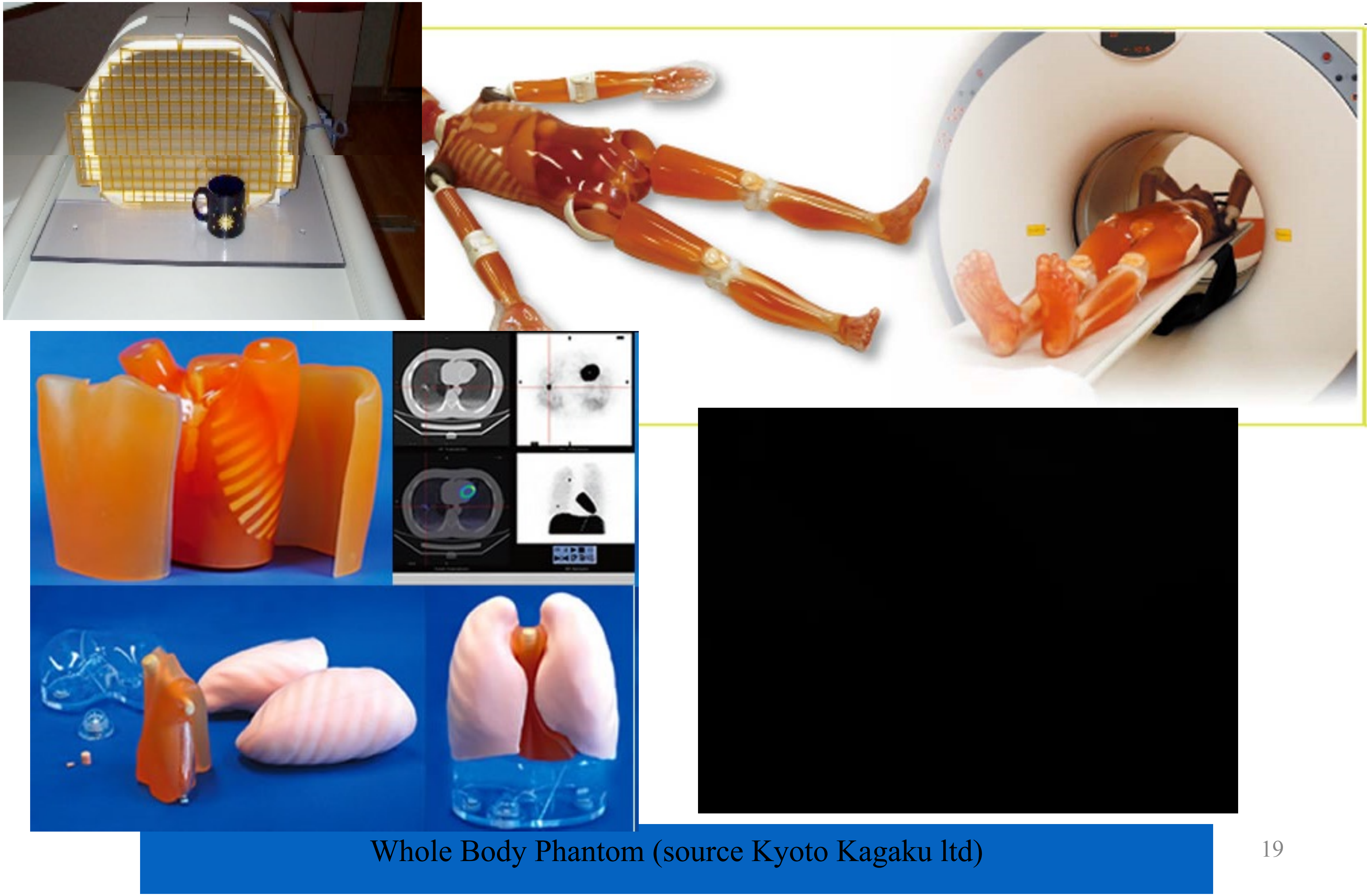
3.7 Variational Bayes EM

3.8 STAPLE Algorithm

Validation of Segmentation Algorithm

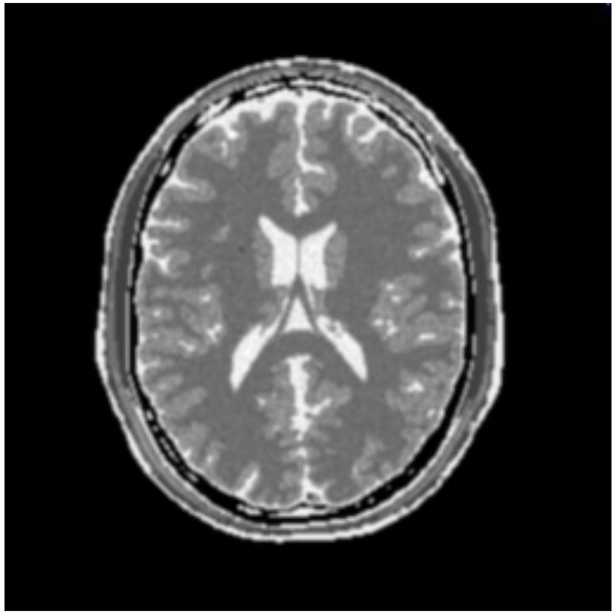
- Intrinsic Validation : comparison against
  - Observation of Physical Phantoms
    - Difficult and expensive to build
    - May not be representative of real data
  - Simulated images (MNI Brain Atlas,…)
    - Difficult to simulate artefacts
  - Segmentation of experts
    - Large inter and intra variability of segmentation across experts
    - May not be representative of population variability

Phantoms for Validation of Segmentation

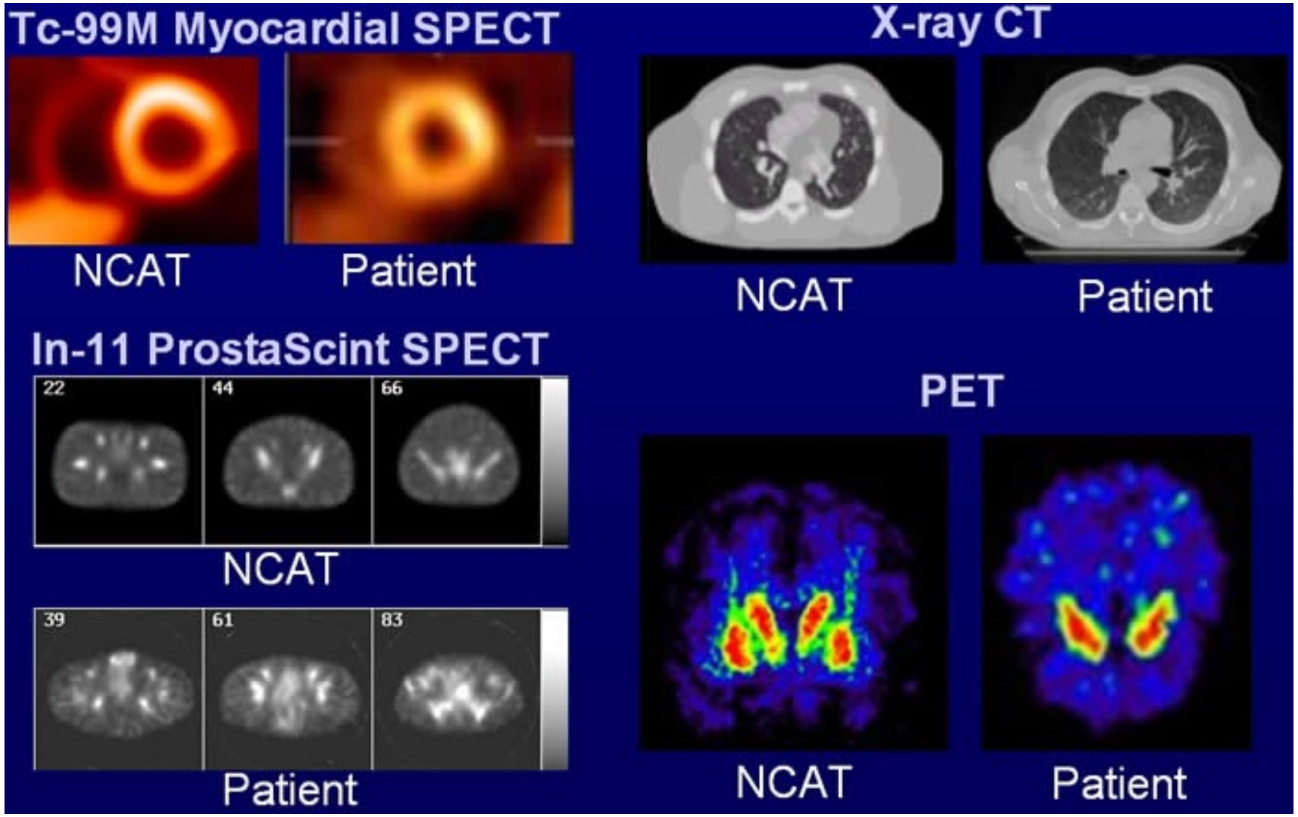


Whole Body Phantom (source Kyoto Kagaku ltd)

Simulation of Medical Images

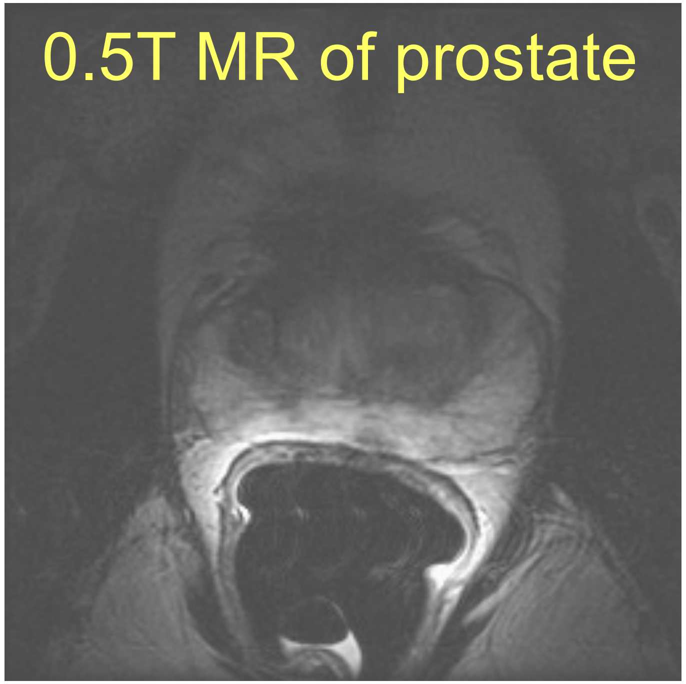


MRI Sim

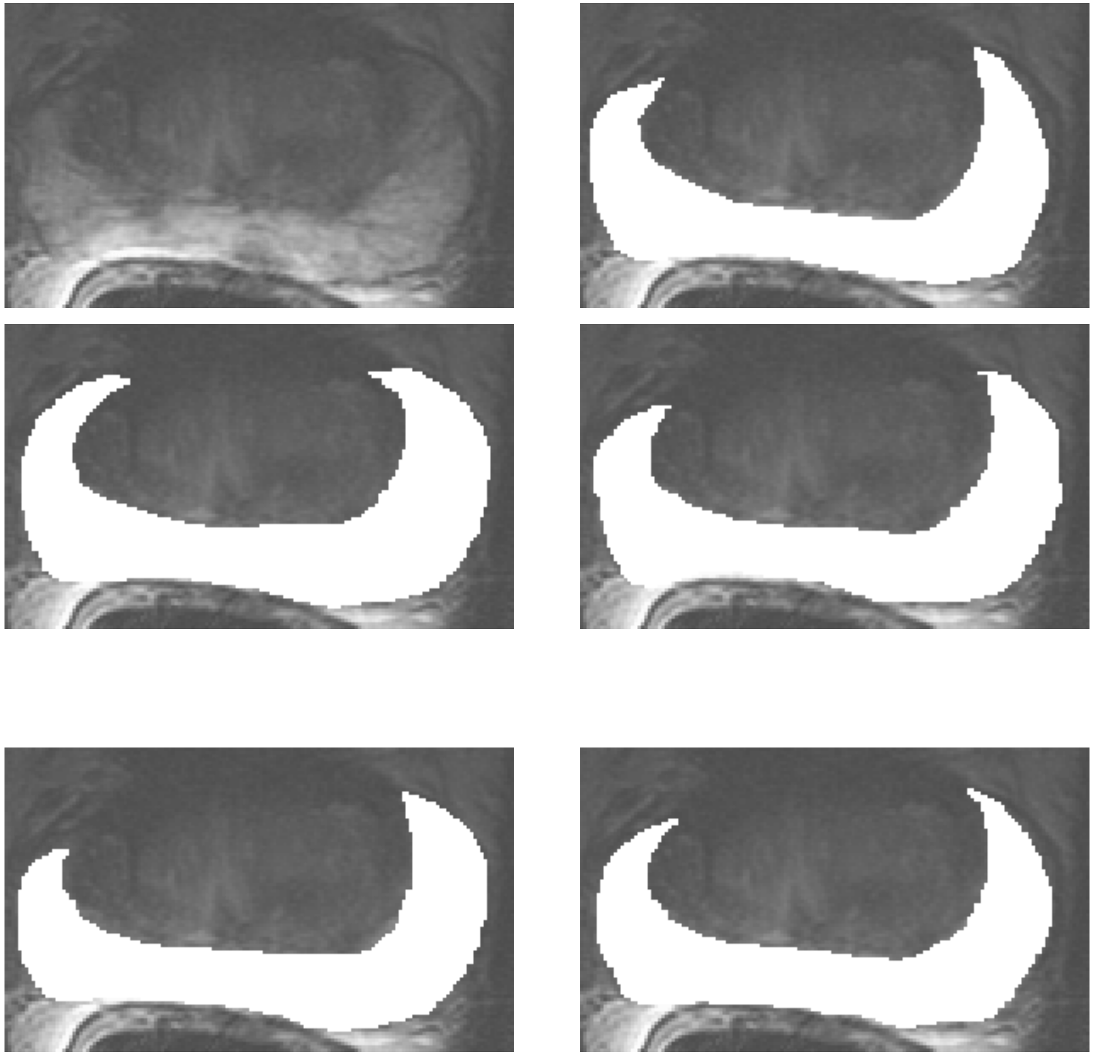


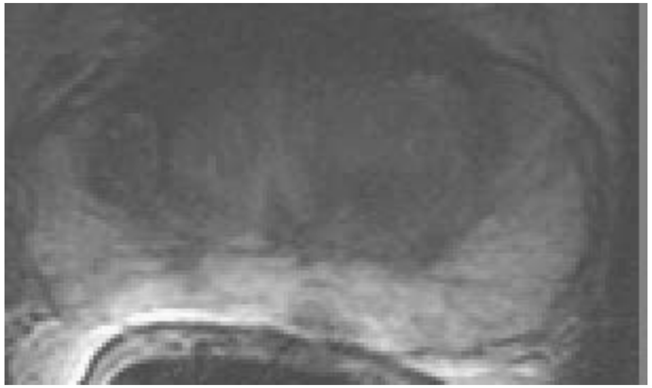
SPECT Image simulation

Segmentation of experts

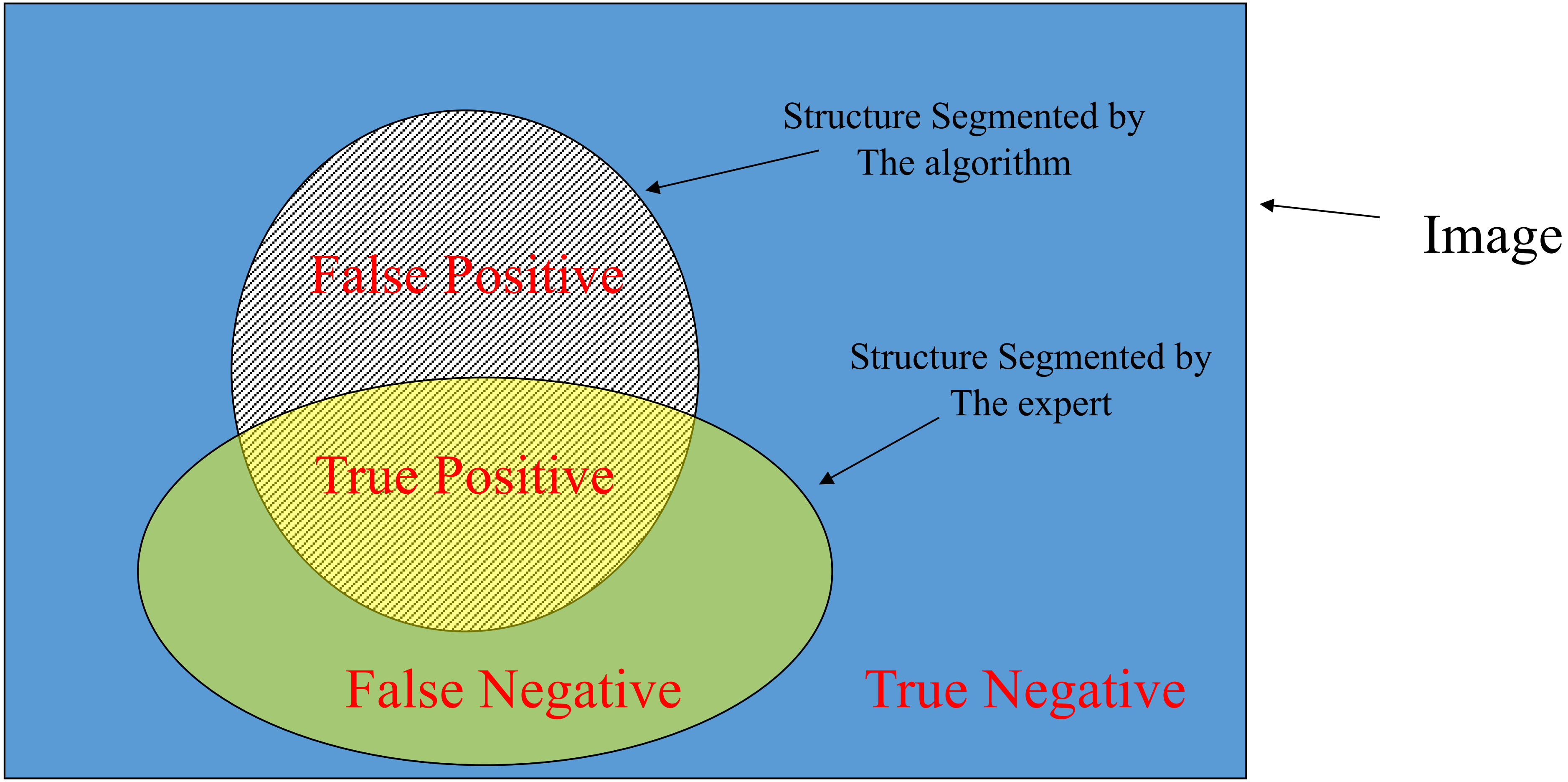


Peripheral zone and segmentations





Measuring the Validity of Segmentation



Measuring the Validity of segmentation

Confusion Matrix

		Expert Segmentation (Ground truth)	
		Present	Absent
Algorithm Segmentation	Segmented = foreground	True positive A	False positive (Type I error) B
	Not segmented= background	False negative (Type II error) C	True negative D

Sensitivity = A / (A+C)

Specificity = D / (B+D)

Sensitivity (or recall): proportion of voxels in the structure which have been segmented by the algorithm

Specificity: proportion of voxels that are not in the structure which have not been segmented by the segmentation algorithm

Measuring the Validity of segmentation

Expert Segmentation (Ground truth)

		Present	Absent
Algorithm Segmentation	Segmented = foreground	True positive A	False positive B
	Not segmented= background	False negative C	True negative D

PPV = A / (A+B)

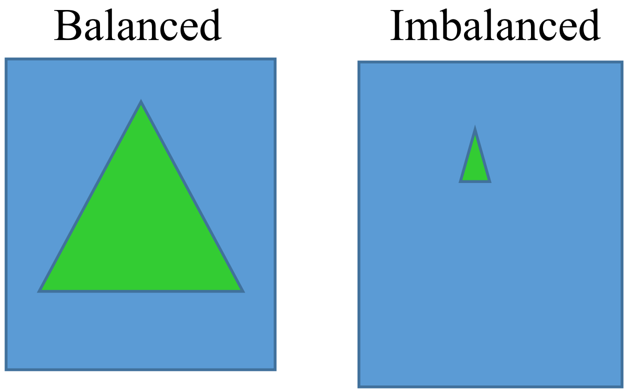
Positive Predictive Value (PPV) or precision : The likelihood that a voxel segmented as foreground is actually a voxel belonging to the structure

NPV = D / (C+D)

Negative Predictive Value (NPV): The likelihood that a voxel not segmented as foreground is actually a background voxel

Measuring the Validity of segmentation

- Often there is an imbalance between foreground and background



- When background >> foreground then specificity and NPV are very close from 1
- Choose metrics independent from background size
  - Sensitivity (recall) and PPV (precision)

Comparing Segmentation Algorithms with ROC Curve (Receiver Operating Characteristic)

Often a segmentation algorithm depends on one parameter (a threshold)

Ideal classifier

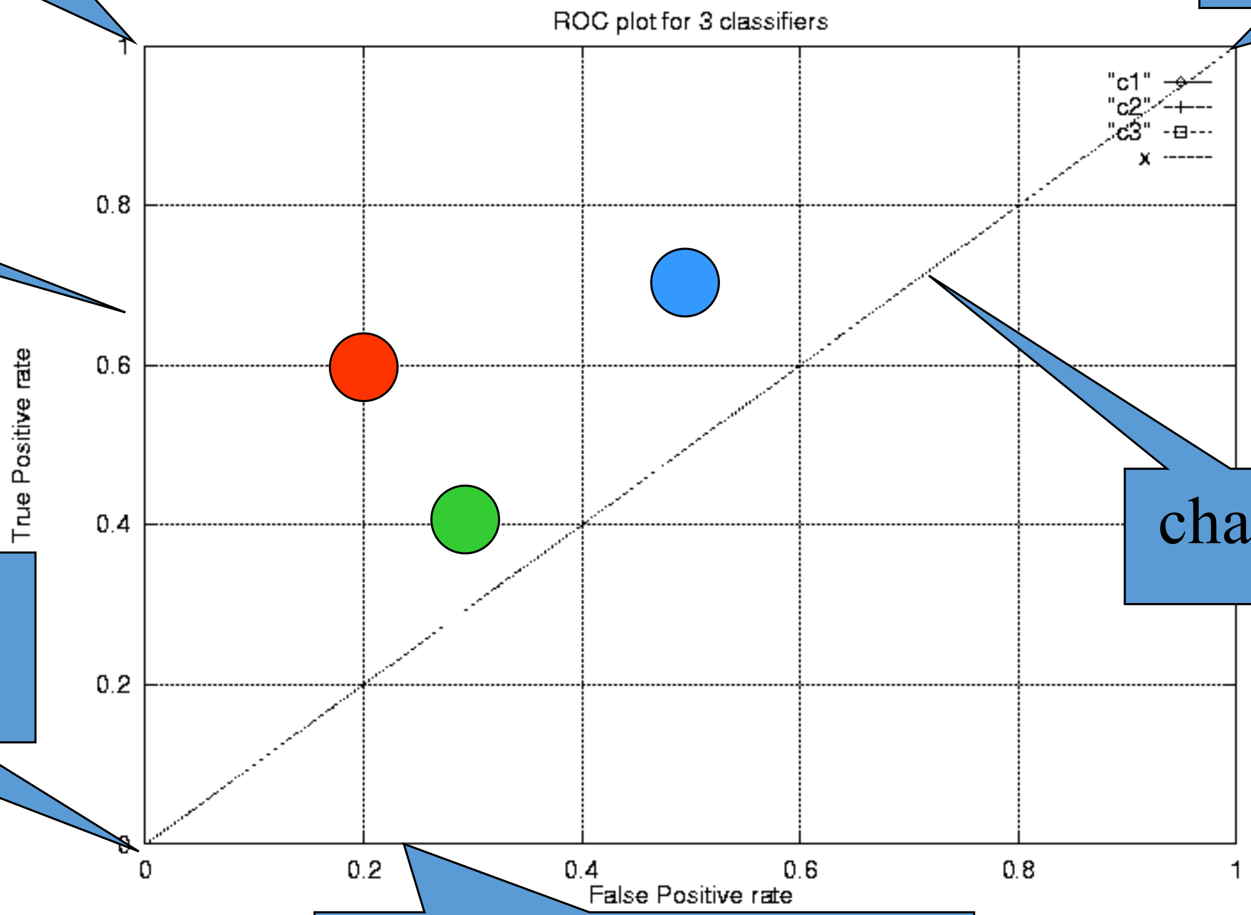
Always positive

Sensitivity

chance

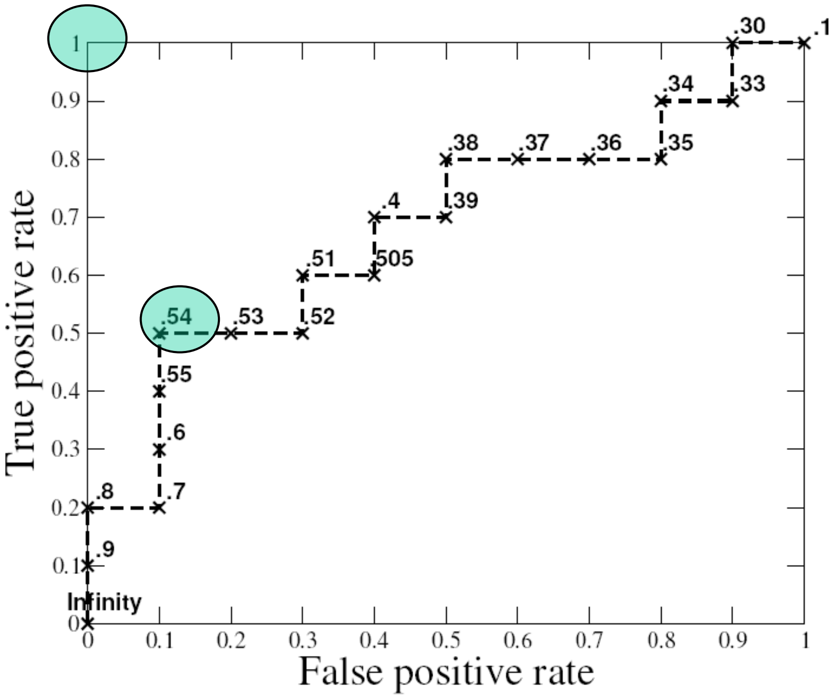
always negative

1-specificity



ROC curves (Receiver Operating Characteristic)

- Use ROC curve to optimize the algorithm
- Pick the value that leads to a point closest from the upper left corner
- Estimate performance of an algorithm by its area under the curve (AUCROC) which is independent from the choice of a threshold



Other measures of segmentation Performance

- Dice Index :

$$s = \frac{2|X \cap Y|}{|X| + |Y|}$$

- Jaccard Index :

$$s = \frac{|X \cap Y|}{|X \cup Y|}$$

X = ground truth
binary object

Y = segmented
binary object

- These are region measures of segmentation performance

- May not be always relevant

$$Dice\ Coefficient = \frac{2 * TP}{FN + (2 * TP) + FP}$$

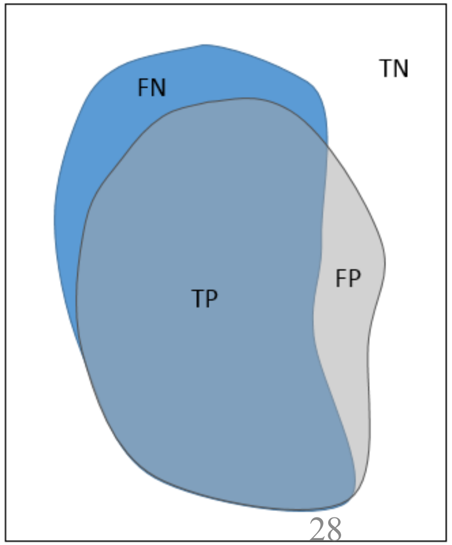
$$Jaccard\ Index = \frac{TP}{TP + FN + FP}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

TP - true positive
TN - true negative
FP - false positive
FN - false negative

Manual Segmentation 🟦
Automated Segmentation ⬜



Boundary measure of segmentation performance

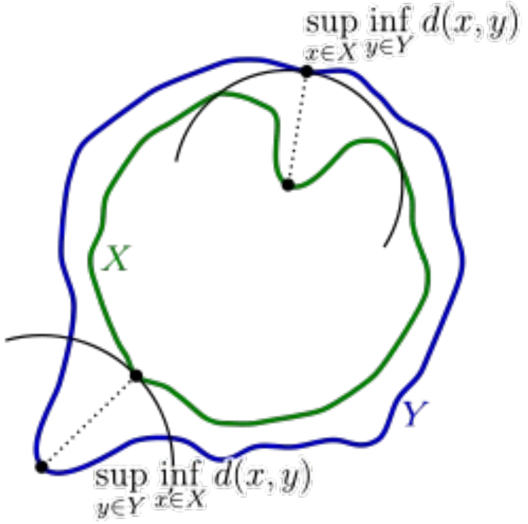
- Hausdorff Distance between surfaces

$$d(X,Y) = Max_{x \in X} Min_{y \in Y} dist(x,y)$$

- Symmetric Hausdorff Distance between surfaces

$$\frac{d(X,Y) + d(Y,X)}{2}$$

- Often consider 95% quantile of (symmetric) Hausdorff distance



Validation of Segmentation Algorithm (2)

- Extrinsic Validation : comparison against other segmentation algorithms
  - Only possible when no ground truth exists (Inter-patient registration of images) or when it is not available
  - Estimate consistency, repeatability and size of convergence basin

3. Medical Image Segmentation

3.1 **Taxonomy of segmentation algorithms**

3.2 Validation of segmentation algorithms

3.3 **Deterministic Filtering & Thresholding Approaches**

3.4 Probabilistic Imaging Model

3.5 Expectation Maximisation for GMM

3.6 Image classification with bias field

3.7 Variational Bayes EM

3.8 STAPLE Algorithm

Thresholding & Mathematical Morphology

- Main Idea :

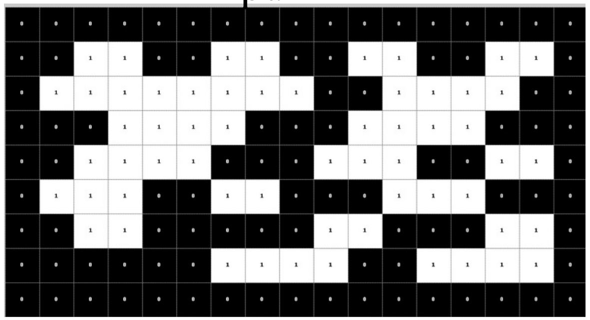
A structure is characterized by its intensity values and its connectivity

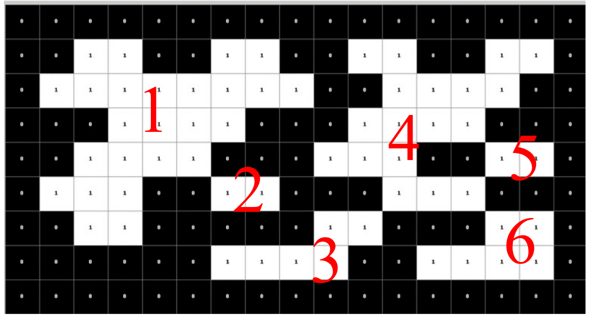
➡ Valid for highly contrasted structures

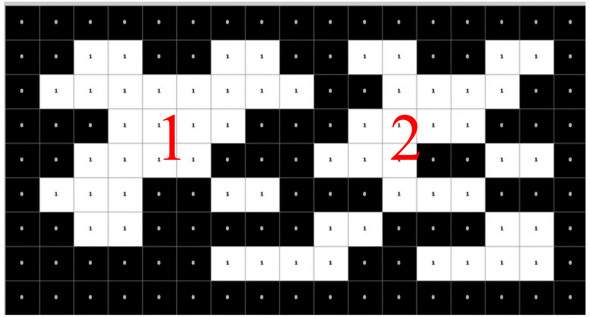
- Basic Algorithm :
  - Thresholding between 2 grey levels (windowing)
  - Mathematical morphology operations
    - Erosion and Dilation
    - Closure & Opening
    - Extraction of connected components

Extraction of Connected Components

- **Input** : a binary image & a choice of neighborhood
- **Output** : for each object voxel provides the index of the connected component to which that voxel belongs


Input

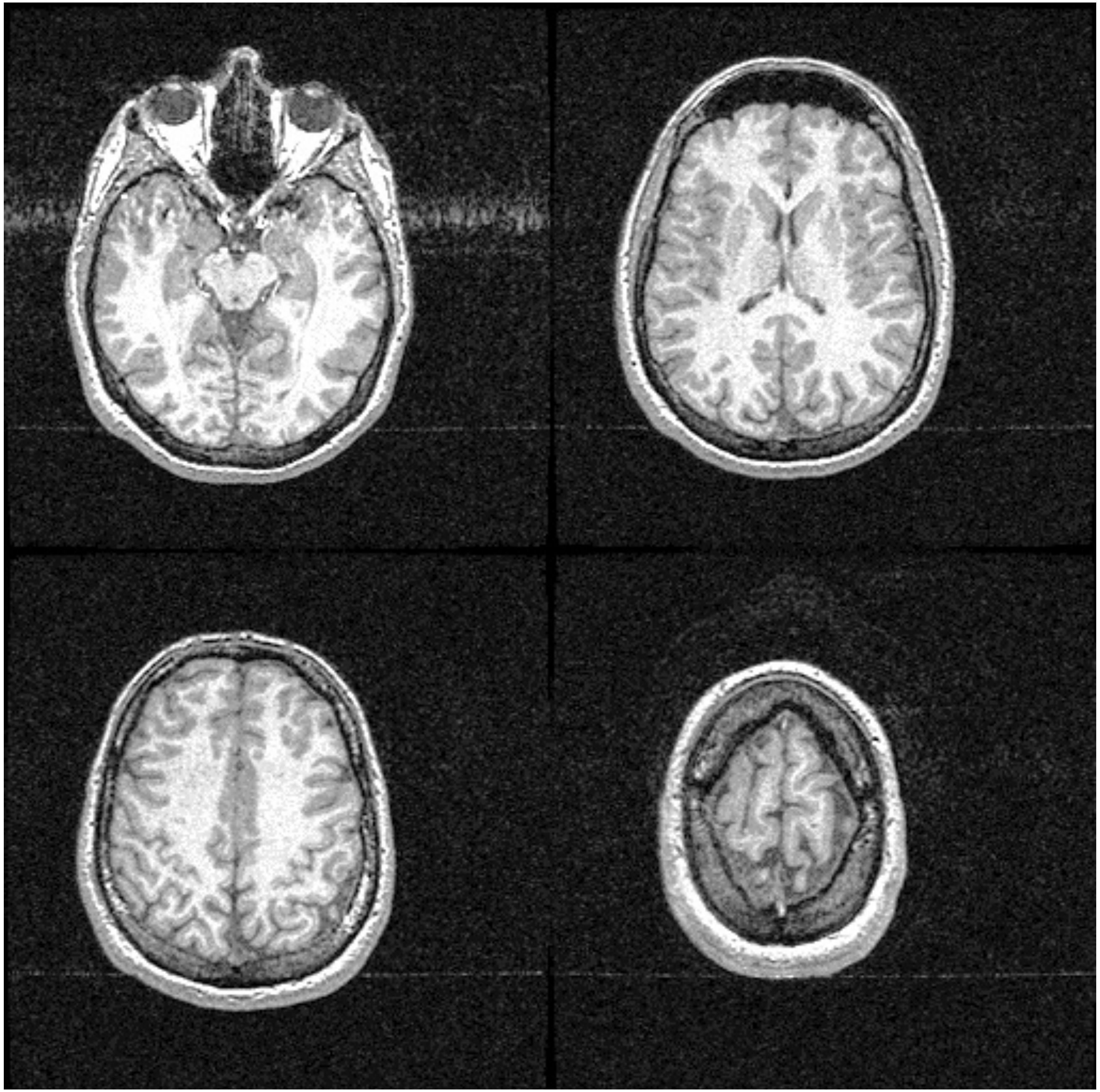

4-connectivity


8-connectivity

- Algorithm performed efficiently in 2 passes
- Often sort components by size

Application

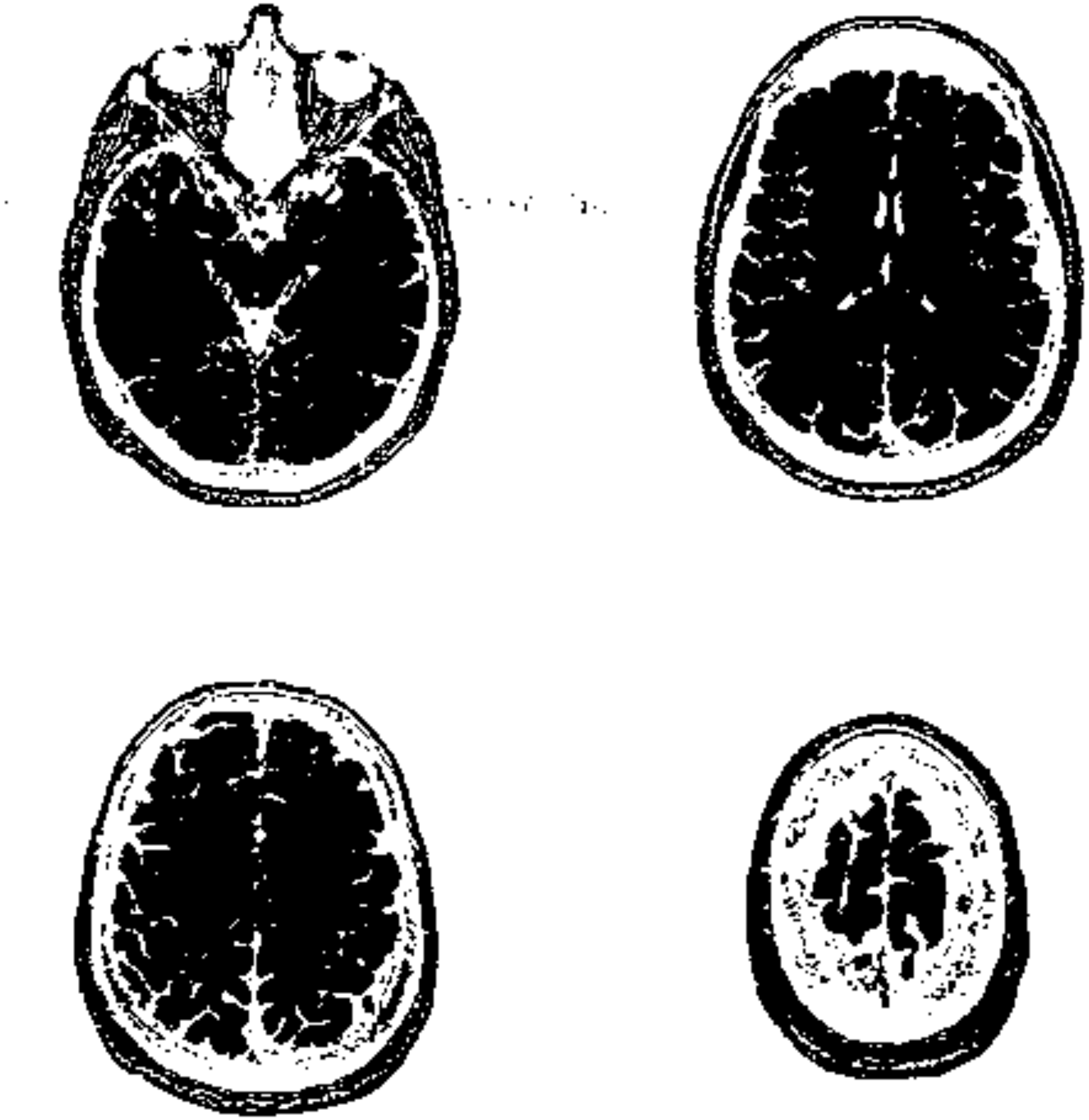
Brain Segmentation of MR Image



Original slices

Application

Brain Segmentation of MR Image

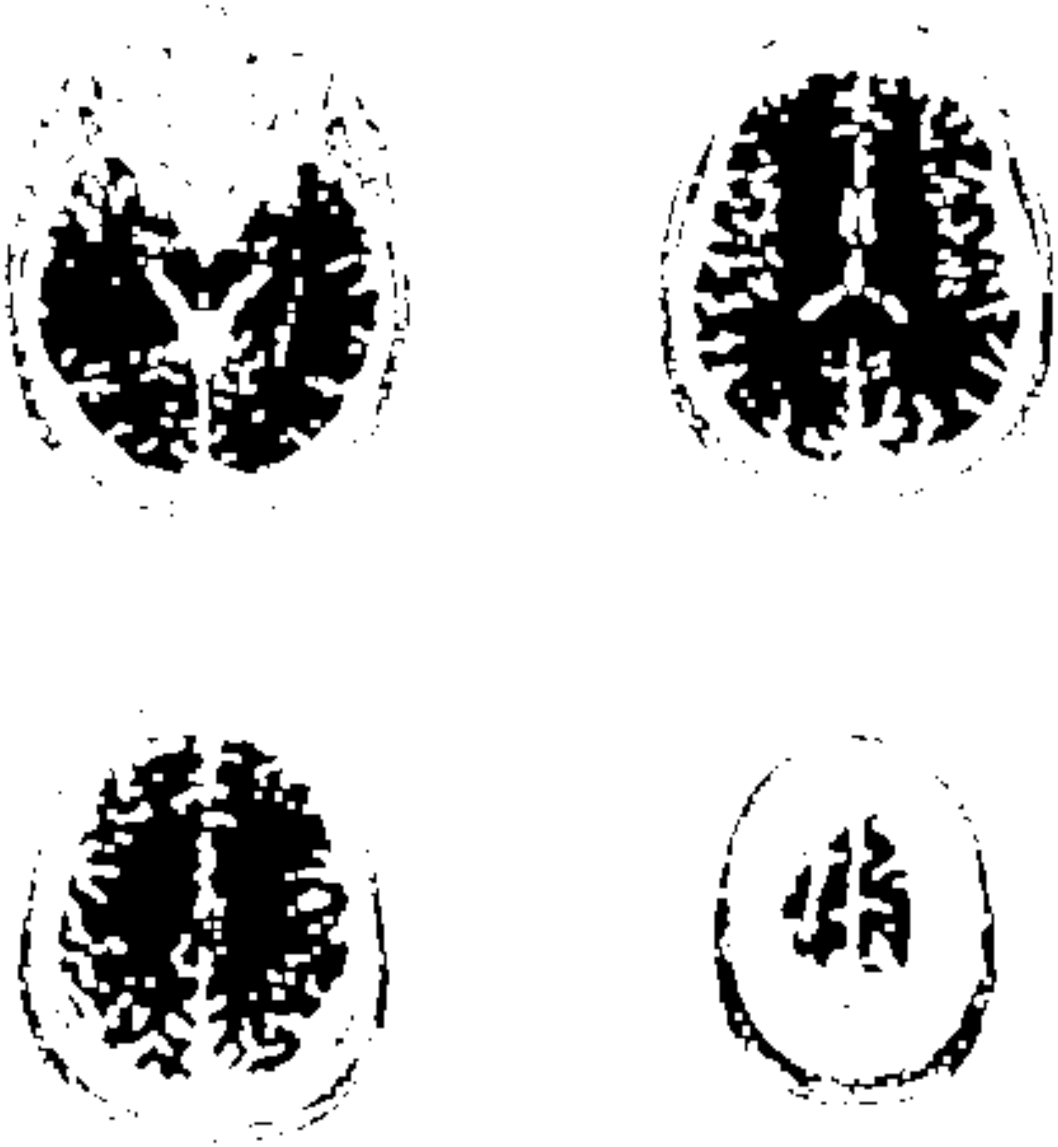


4 slices after thresholding

Application

Brain Segmentation of MR Image

4 slices after a single 3D erosion



Application

Brain Segmentation of MR Image

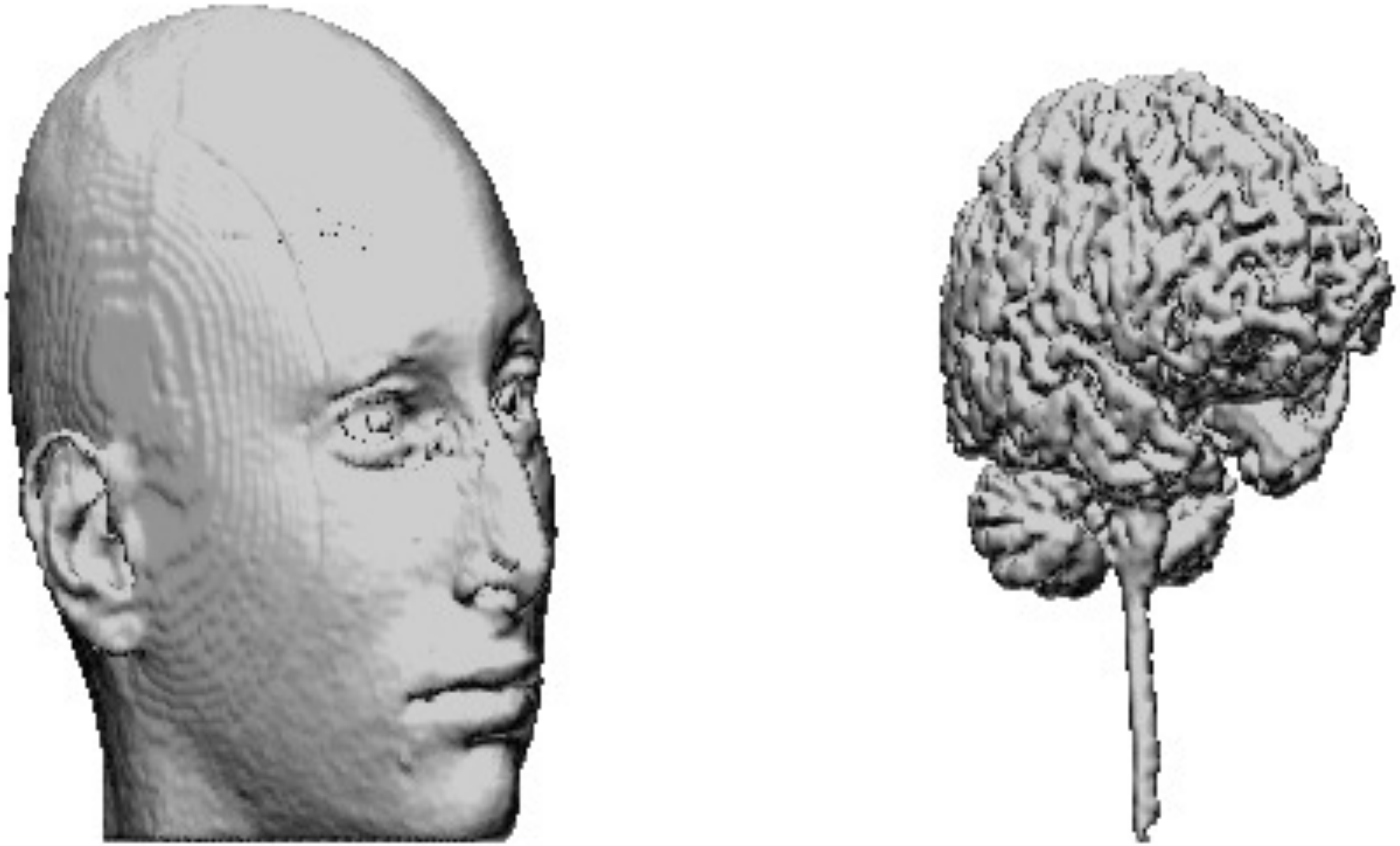
4 slices after extraction of the largest connected component

Application

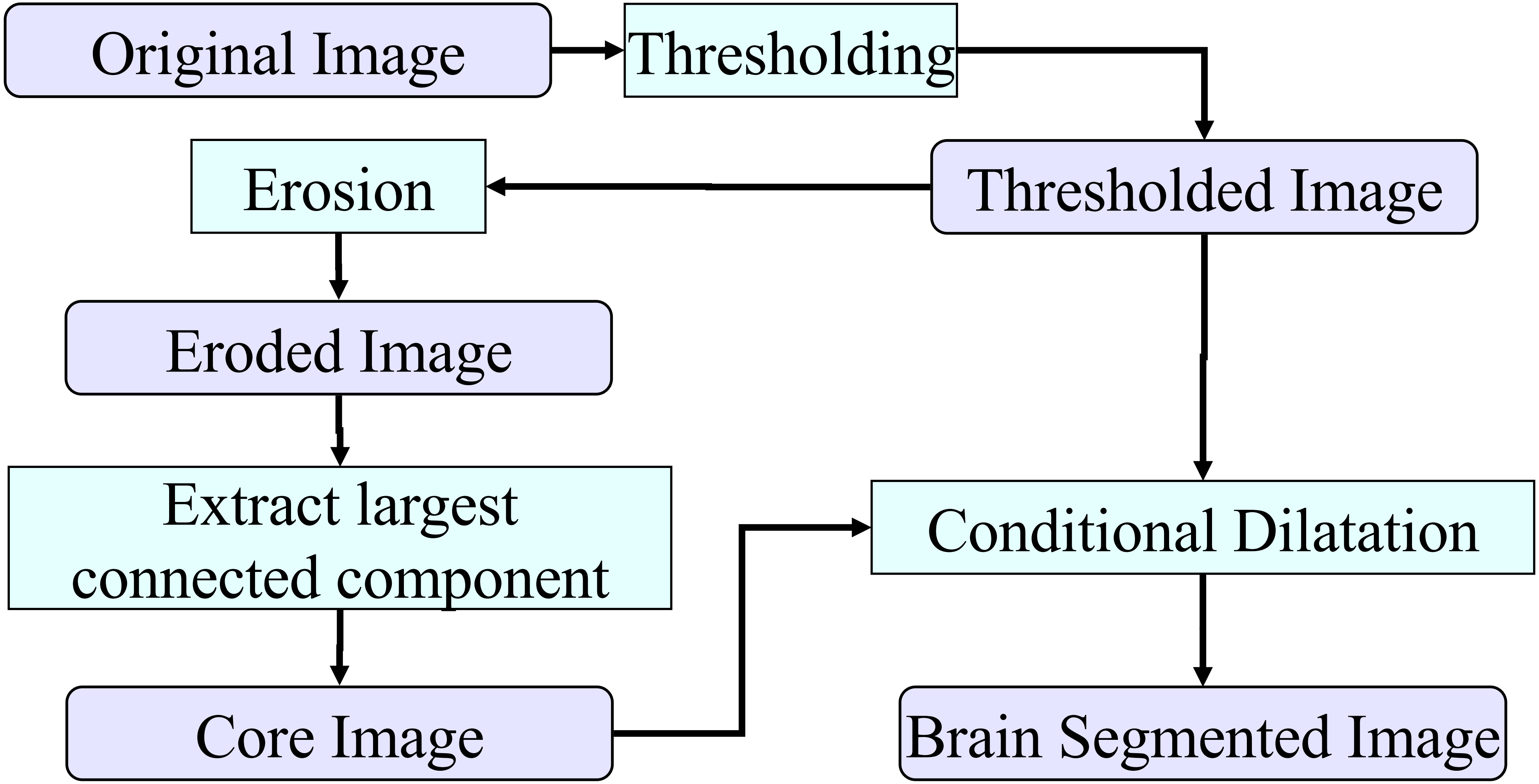
Brain Segmentation of MR Image

4 slices after 3D conditional dilation

Application



Brain Extraction



Original Image

Thresholding

Thresholded Image

Erosion

Eroded Image

Extract largest connected component

Core Image

Conditional Dilatation

Brain Segmented Image

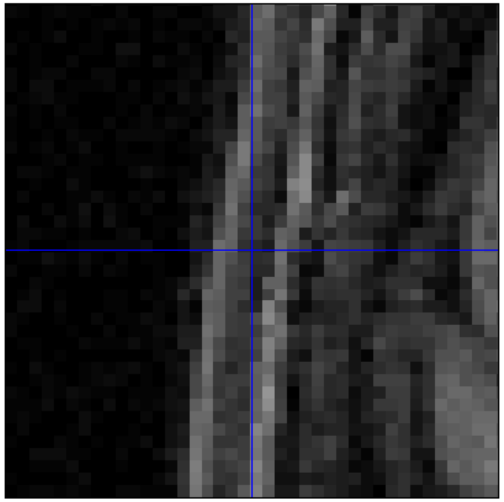
Limitations of Thresholding

- Difficulty to select threshold, e.g. from grey-level histogram (Otsu’s method)
- Create staircase effects since assignment of one voxel to one class
  - Does not take into account the effect of **partial volume effect** (PVE)
- Does not assume any spatial correlation of voxel intensity (isolated voxels)

→ Use of classification methods

Interest of Image Classification

Noise & Partial volume effect

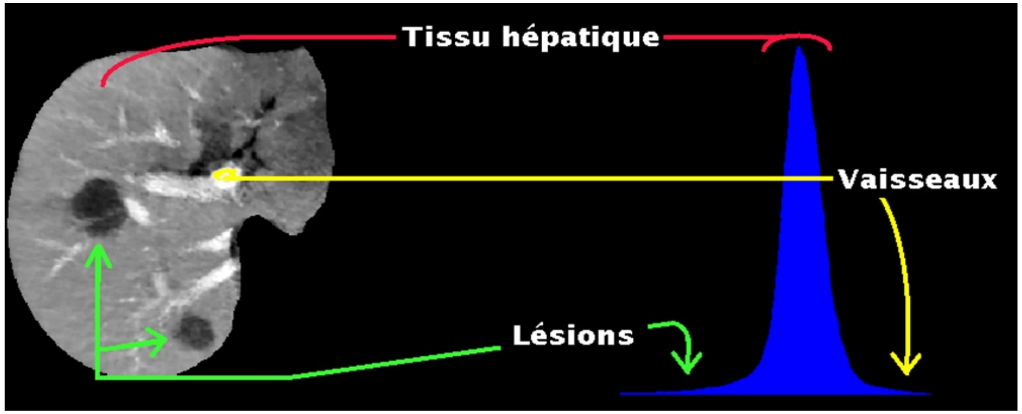


Brain MRI

Liver from

CT image with 3 classes :

Lesion, vessels & parenchyma



MR Bias Field

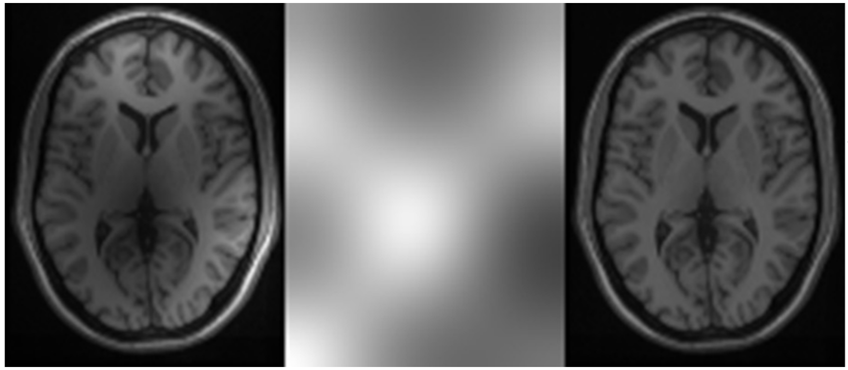


Image with bias artefact

Corrected image

3. Medical Image Segmentation

- 3.1 Taxonomy of segmentation algorithms
- 3.2 Validation of segmentation algorithms
- 3.3 Deterministic Filtering & Thresholding Approaches
- 3.4 **Probabilistic Imaging Model**
- 3.5 Expectation Maximisation for GMM
- 3.6 Image classification with bias field
- 3.7 Variational Bayes EM
- 3.8 STAPLE Algorithm

Probability Reminder

- Conditional Probability

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A,B)}{P(B)}$$

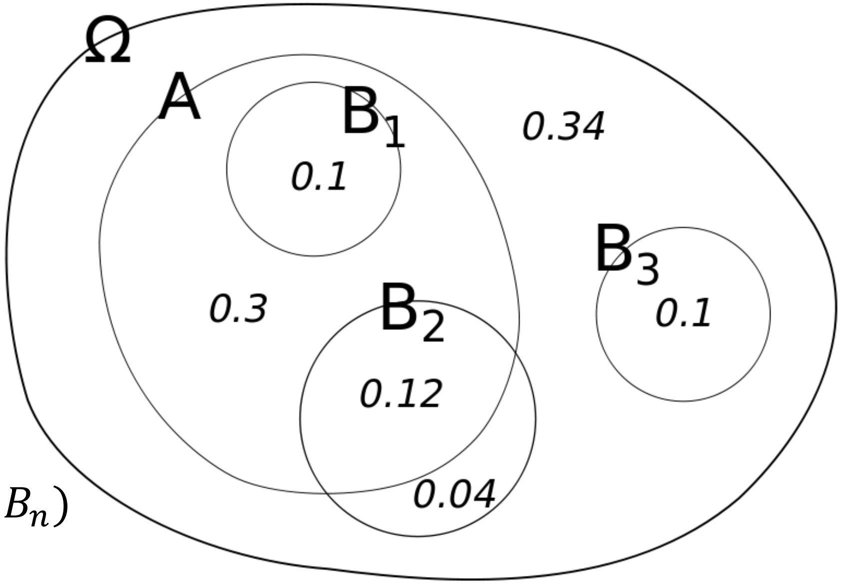
- Total Probability

discrete

$$P(A) = \sum_n P(A \cap B_n) = \sum_n P(A|B_n)P(B_n) = \sum_n P(A, B_n)$$

continuous

$$p(A) = \int_B p(A|B_x\,)p(B_x)dB_x$$



- Bayes Law

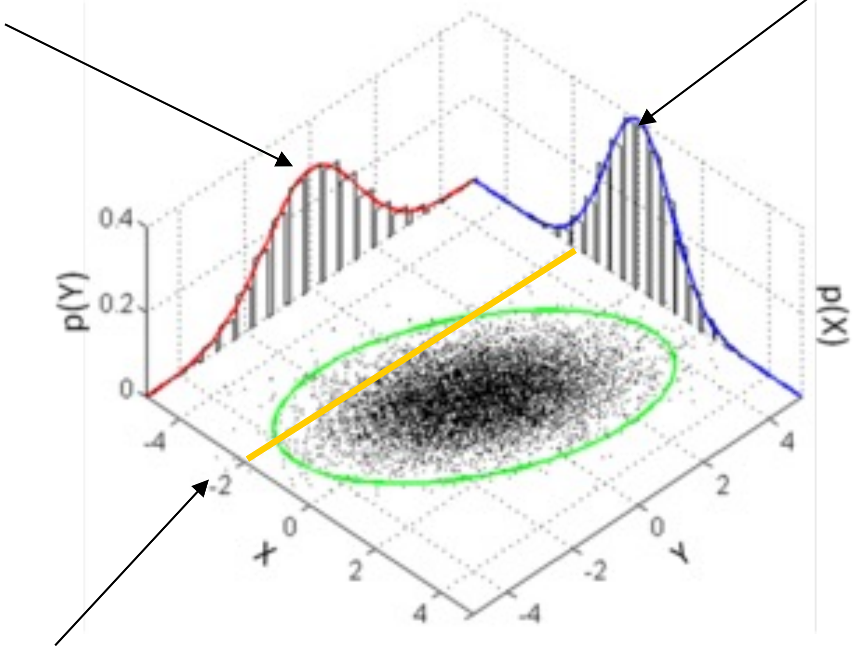
$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Conditional & Marginal probability

- Distribution of a pair (x,y) of random variables

Marginal Probability $p(y) = \int p(x,y)\, dx$

Marginal Probability $p(x) = \int p(x,y) dy$



Conditional Probability p(y|x= -2)

Distance between distributions

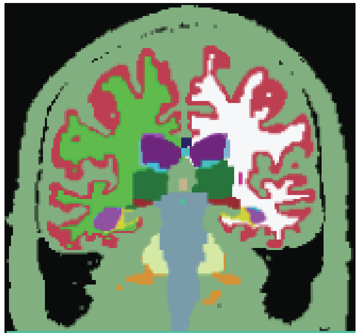
- How similar are 2 probability distribution functions ?
  - Kullback-Leibler Divergence or relative entropy: $D_{kL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$
    - Non symmetric
    - Always positive
    - Null iff the two distributions are equal
  - Hellinger distance

$$D_H(P||Q)^2 = \frac{1}{2} \sum_i \left(\sqrt{P(i)} - \sqrt{Q(i)}\right)^2$$

Generic Probabilistic Imaging Model

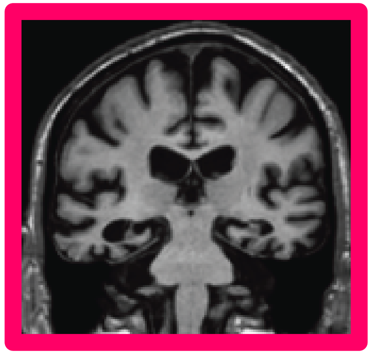
Labelling Process

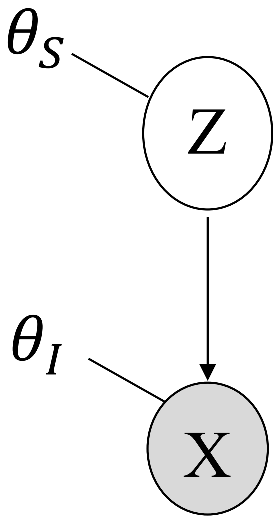
$p(Z|\theta_S)$



Imaging Process

$p(X|Z,\theta_I)$





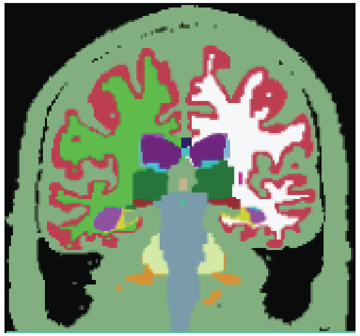
Fundamental assumption : Image intensities depends on voxel class

Parameters θ_S and θ_I may be parameters or random variables and are unknown

Generic Probabilistic Imaging Model

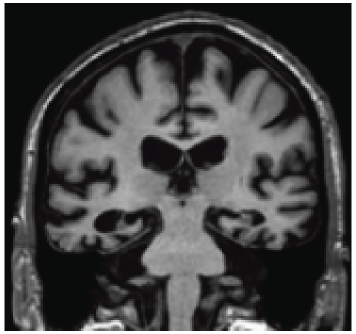
Labelling Process

$p(Z|\theta_S)$



Imaging Process

$p(X|Z,\theta_I)$



Notations :

- z_n label of voxel n
  One in K coding $z_{nk} = 1$ if voxel n belongs to class k
  z_n is a vector of K binary variables
- $Z = \{z_n\}$ set of image labels
- θ_S set of label parameters (e.g. atlas related parameters, shape parameters)

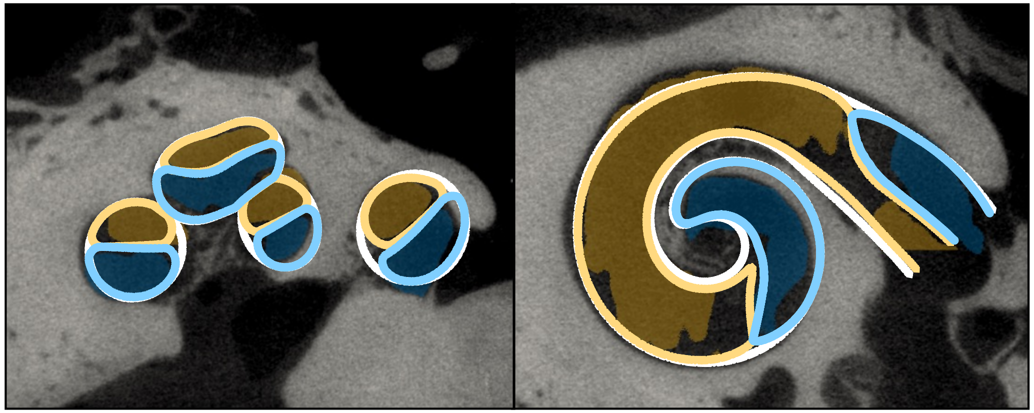
Notations :

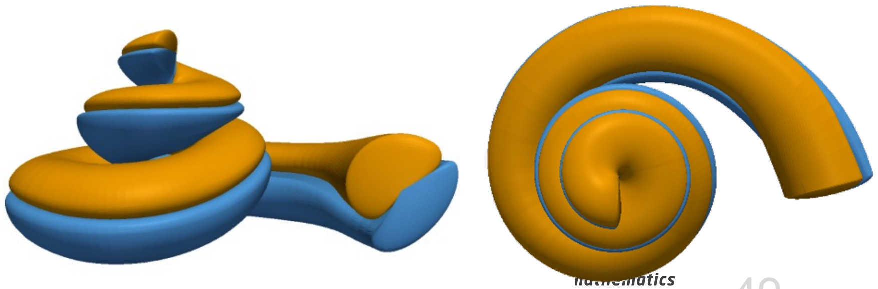
- x_n intensity vector of voxel n of dimension d
- $X = \{x_n\}$ set of image intensities
- θ_I set of imaging parameters (e.g. Gaussian mixture parameters)

Hypothesis on Labelling Process

- Prior $p(Z|\theta_S)$ $z_n = \begin{bmatrix} z_{n1} \\ \cdots \\ z_{nk} \end{bmatrix}$
- By construction $\sum_k p(z_{nk} = 1) = \sum_k z_{nk} = 1$
- Common choices :
  - Random labeling : uninformative prior
  - Homogeneous prior (same probability for all voxels):

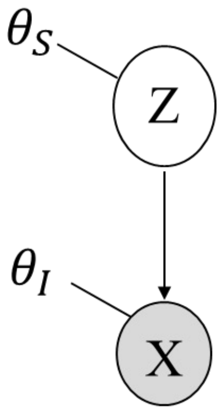
    $p(z_{nk} = 1) = p(z_{mk} = 1) = \pi_k$ $p(z_n) = \sum_k z_{nk}\pi_k = z_n \cdot \pi$
  - Labels from Atlas registration
  - Labels from parametric model





Segmentation Problem as Maximization of probability

- Segmentation is **an inverse problem** consisting in estimating labels Z , and parameters θ_i and θ_S from the knowledge of Intensities X
- Posterior probability :
  - $p(Z|X,\theta_S,\theta_I) = \frac{p(X|Z,\theta_I)p(Z|\theta_S)}{p(X|\theta_I,\theta_S)}$ through Bayes Law



- Likelihood :
  - *Likelihood* $p(X|Z,\theta_I)$ is the probability of observing the data given the label and image parameters
- Marginal likelihood or Evidence :
  - $p(X|\theta_I,\theta_S) = \sum_Z p(X|Z,\theta_I)p(Z|\theta_S)$ is a) is only a function of parameters θ_i and θ_S thus suitable for i) optimization of parameters and ii) for model selection.
  - It is often untractable but can be approximated by a lower bound

Segmentation Problems

- Hard Segmentation :
  - Objective is to estimate Z, i.e provide one label per voxel
- Soft Segmentation (aka classification)
  - Objective is to estimate posterior probability of each voxel $p(z_{nk} = 1|X)$ such that they sum to 1

$$p(Z|X, \theta_S, \theta_I) = \frac{p(X|Z, \theta_I)p(Z|\theta_S)}{\sum_{Z^*} p(X|Z^*, \theta_I)p(Z^*|\theta_S)}$$

  - Require estimating parameters θ_I and θ_S

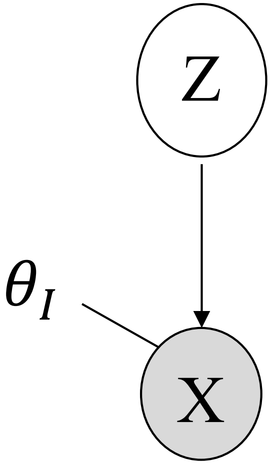
Inference approaches

- Point estimates :
  - Maximum Likelihood
  - Maximum a posteriori
- Posterior estimates
  - Exact inference
  - Variational Bayes
  - Stochastic Sampling

For random variables
& parameters

For random variables
only

Maximum Likelihood



- Can be used for parameters or random variable

$$\widehat{\theta_I} = \arg\max_{\theta_I} p(X|Z,\theta_I)$$

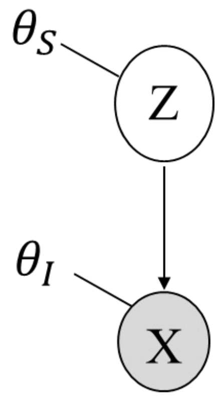
Maximum Likelihood
For image parameters

$$\hat{Z} = \arg\max_{Z} p(X|Z,\theta_I)$$

Maximum Likelihood
For label

$$(\widehat{\theta_I},\hat{Z}) = \arg\max_{Z,\theta_I} p(X|Z,\theta_I)$$

Maximum a posteriori



- Maximize posterior probability or joint probability
- Bayes Law :$p(Z|X,\theta_S,\theta_I) = \frac{p(X|Z,\theta_I)p(Z|\theta_S)}{p(X)} = \frac{p(X,Z|\theta_I,\theta_S)}{p(X)}$

$$\boxed{(\hat{Z},\widehat{\theta_I},\widehat{\theta_S}) = \arg\max_{Z,\theta_I,\theta_S} \underbrace{p(X|Z,\theta_I)}_{\text{Imaging Term}}\ \underbrace{p(Z|\theta_S)}_{\text{Label Term}}}$$

Maximum A Posteriori

- If labels and intensity are independent $(p(Z) = \prod_n p(z_n))$, $(p(X) = \prod_n p(x_n))$ then equivalent to assigning a label to each voxel

Posterior estimates

- Exact posterior in simple cases
  $p(Z|X,\theta_I,\theta_S)$, $p(\theta_I|X)$, $p(\theta_S|Z)$
- Approximate posterior distribution :
  - Variational Bayes : seek $q(Z)$ as approximation of $p(Z|X,\theta_I,\theta_S)$ which minimized $D_{KL}(p(Z|X,\theta_I,\theta_S)||q(Z))$
  - Stochastic sampling (e.g. Gibbs sampling, MCMC)

General Taxonomy of methods

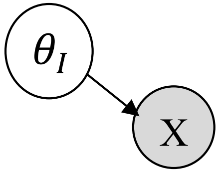
Combination of difference inference methods for different variables

	Label Z	Parameter θ_I	Parameter θ_S
Maximum likelihood			
Maximum a posteriori			
Exact Posterior			
Variational Bayes			
Stochastic Sampling			

Notation Reminder

- K is the number of classes : $K \geq 1$
- N is the number of voxels
- d is the dimension of the feature vector x_n
- $p(z_{nk} = 1|\theta_S)$ is the **prior** on the label of class k at voxel n
- $p(z_{nk} = 1|x_n)$ is the **posterior** probability of having label k at voxel n
- $p(x_n, z_{nk})$ is the **joint probability** of having voxel intensity x_n and label k
- $p(x_{nk} = 1|\theta_I, \theta_S)$ is the **marginal likelihood**
- $p(x_{nk} = 1|\theta_I, Z_n\ )$ is the **likelihood**

Example 1 : Multivariate Gaussian Image



- Hypothesis :
  - **All voxels are independent** : $p(X|\theta_I) = \prod_n p(x_n|\theta_I)$ and $p(Z) = \prod_n p(z_n)$=1
  - **Only one class** K=1 !! Everywhere $z_{n1} = 1$
  - Voxel intensities $\boldsymbol{x_n}$ **are vectors** :
    - For instance intensity, gradient, second derivatives
    - Multi sequence MR images : PD, T1, T2, Flair
  - $\boldsymbol{p(x_n|\theta_I)}$ **is a multivariate Gaussian**

Gaussian Distribution

- We assume that for a given class of tissue k, the intensity follows a Gaussian Distribution

$$P(I|\mu_k,\sigma_k) = \mathcal{N}(I|\mu_k,\sigma_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}}\exp\left(-\frac{(I-\mu_k)^2}{2\sigma_k^2}\right)$$

mean

Standard deviation

Multivariate Gaussian

- We suppose that at each voxel there is a feature vector x of size d
- Introduce mean vector μ, covariance matrix Σ as $d \times d$ positive definite matrix

$$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^t\,\Sigma^{-1}(x-\mu)\right)$$

Mean Vector

Covariance Matrix

Determinant of Covariance Matrix

Example 1 : Maximum Likelihood

- For multivariate Gaussian $\theta_I = \{\mu \quad , \Sigma \quad \}$
- Objective : given image X, estimate mean μ and covariance Σ
- Equivalently maximize the log likelihood

$$\ln p(X|\mu, \Sigma) = -\frac{N}{2}\ln|\Sigma| - \frac{dN}{2}\ln(2\pi) - \frac{1}{2}\sum_{n=1}^{N}(x_n - \mu)^T\Sigma^{-1}(x_n - \mu)$$

Vector and Matrix Derivation

- For any vector x

$$\frac{\partial\left(\frac{x^T Ax}{2}\right)}{\partial x} = Ax$$

- For any matrix A

$$\frac{\partial \ln|A|}{\partial A} = \left(A^{-1}\right)^T = A^{-T}$$

- For symmetric Matrix A

$$\frac{\partial\left(x^T A^{-1} y\right)}{\partial A} = -\mathrm{A}^{-T} x y^T \mathrm{A}^{-T}$$

Maximum Likelihood Solution

- Maximizing w.r.t. the mean gives the *sample mean*

$$\mu_{ML} = \frac{1}{N}\sum_{n=1}^{N} x_n$$

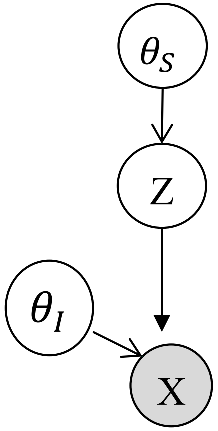
Maximizing w.r.t covariance gives the *sample covariance*

$$\Sigma_{ML} = \frac{1}{N}\sum_{n=1}^{N}(x_n - \mu_{ML})(x_n - \mu_{ML})^T$$

Example 2 : Gaussian Mixture

- Hypothesis :
  - All voxels are independent : $p(X|Z) = \prod_n p(x_n|z_n)$ and $p(Z) = \prod_n p(z_n)$
  - More than one class $K \geq 2$
  - Voxel intensities x_n are vectors
  - Label Priors are unknown but homogeneous :

    $$\forall n, m \; p(z_{nk} = 1) = p(z_{mk} = 1) = \pi_k$$
  - $p(x_n|z_{nk} = 1) = \mathcal{N}(x_n|\theta_k)$ is a multivariate Gaussian



- Notations : $\theta_k = \{\mu_k, \Sigma_k\}$ $\theta \quad = \{\theta_k, \pi_k\}$

$$\theta_S = \{\pi_k\} \; \theta_I = \{\theta_k\}$$

Gaussian Mixture

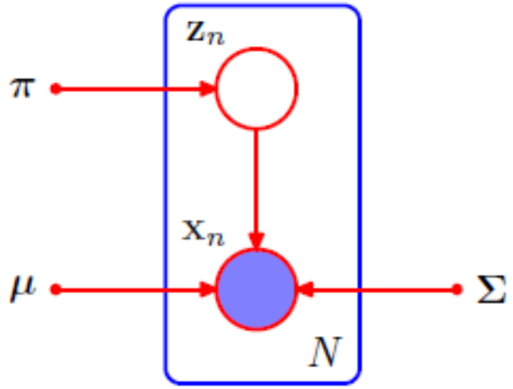
- $p(x_n|z_n) = \sum_k z_{nk} \mathcal{N}(x_n|\theta_k)$
- Marginal likelihood obtained by law of total probability

$$\mathrm{p}(\mathrm{x}_\mathrm{n}) = \sum_{z_n} \mathrm{p}(x_n|z_n)p(z_n) = \sum_k \pi_k \mathcal{N}(x_n|\theta_k)$$

- Mixing coefficients π_k are homogeneous

$$\sum_{k=1}^{K} \pi_k = 1 \quad 0 \leq \pi_k \leq 1$$

Joint Probability



- Graphical model which reflects the fact that Z explains X

- Define the joint probability

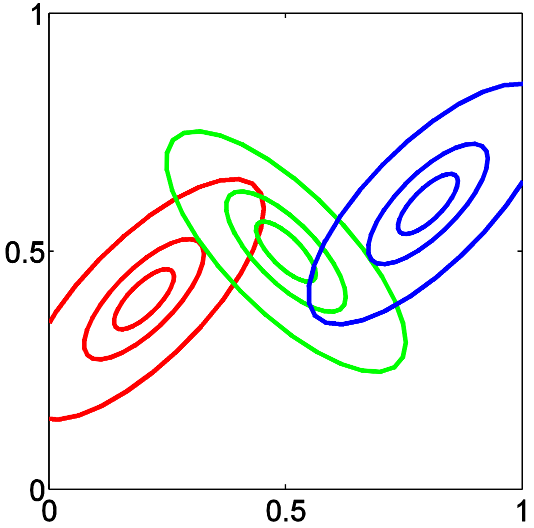
$$p(x_n, z_{nk} = 1) = p(x_n | z_{nk} = 1) p(z_{nk} = 1) = \pi_k \mathcal{N}(x_n | \theta_k)$$

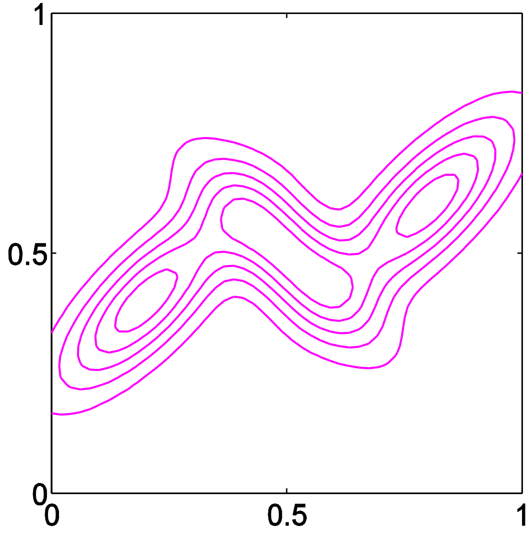
$$p(x_n, z_n) = \sum_k z_{nk}\, \pi_k \mathcal{N}(x_n | \theta_k)$$

Sampling a Gaussian Mixture

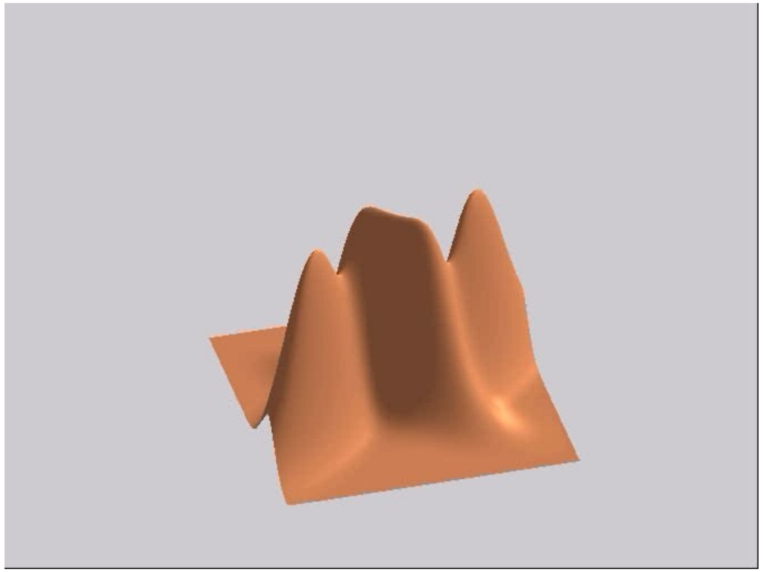
- To generate a data point:
  - first pick one of the components with probability π_k
  - then draw a sample x_i from that component following Gaussian law with parameter θ_k
- Repeat these two steps for each new data point

Mixture of 3 Gaussians



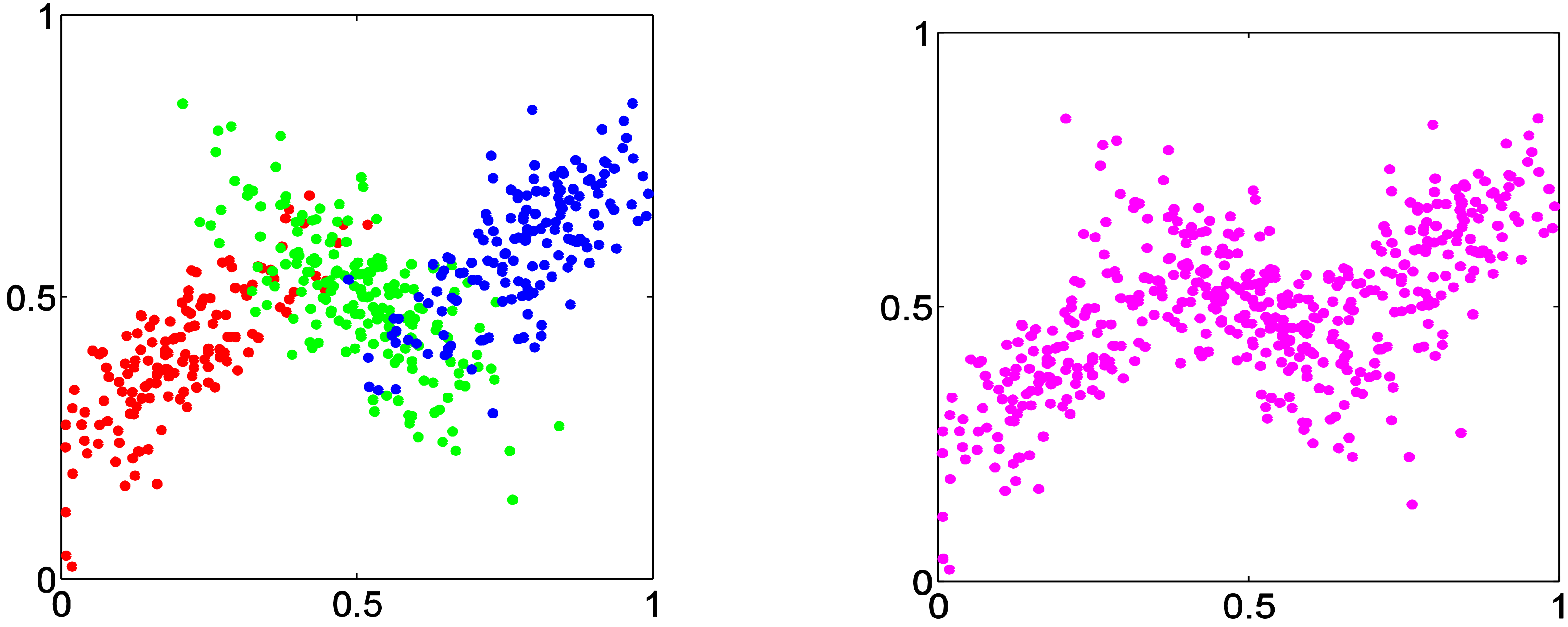


Contour of probability distribution



Surface Plot

Sampled Gaussian Mixture



Gaussian Mixtures & posterior probabilities

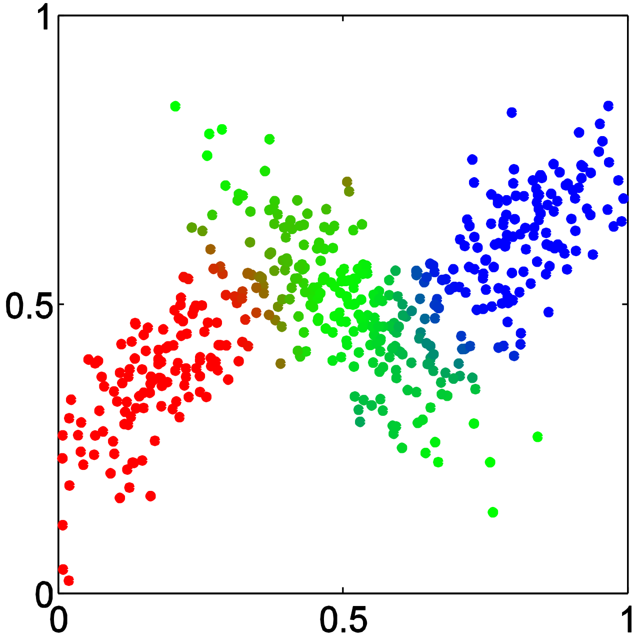
- Marginal Likelihood

$$\mathrm{p(x)} = \sum_{z} p(x|z)p(z) = \sum_{k=1} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$$

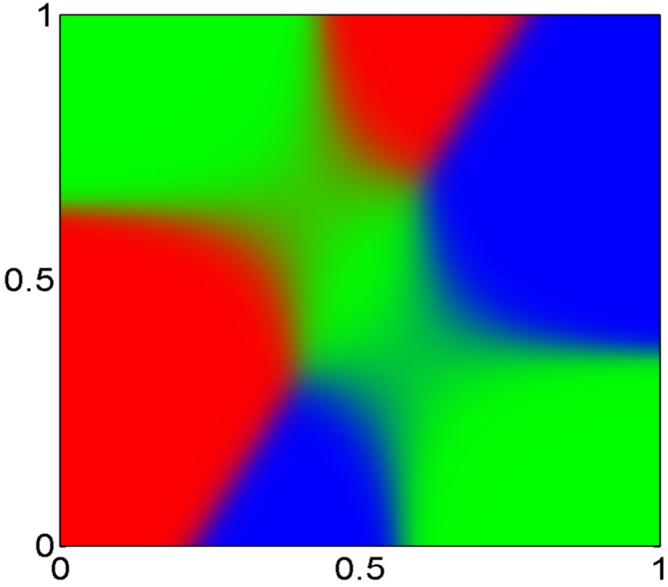
- Use Bayes law to obtain the posterior probabilities

$$p(z_{nk} = 1|x_n) = \frac{p(x_n|z_{nk} = 1)p(z_{nk} = 1)}{p(x_n)} = \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \, \mathcal{N}\left(x_n|\mu_j, \Sigma_j\right)}$$

Posterior Probabilities (colour coded)



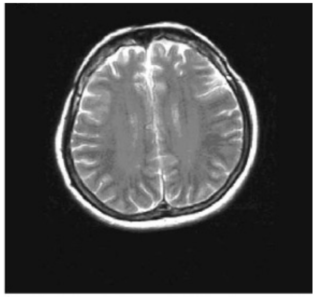
Posterior Probability Map



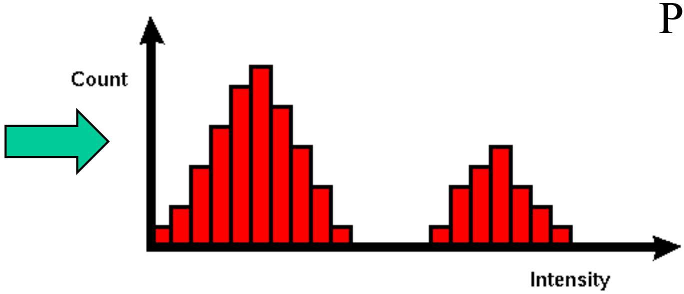
Dense Posterior Probability Map

Gaussian Mixture & Images

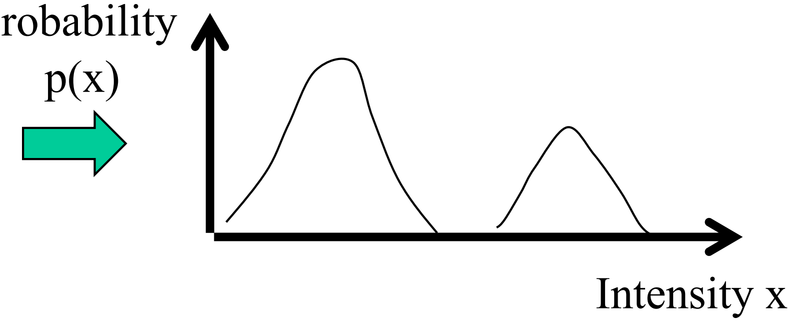
- If x is the image intensity then p(x) can be estimated with the normalized histogram



Image



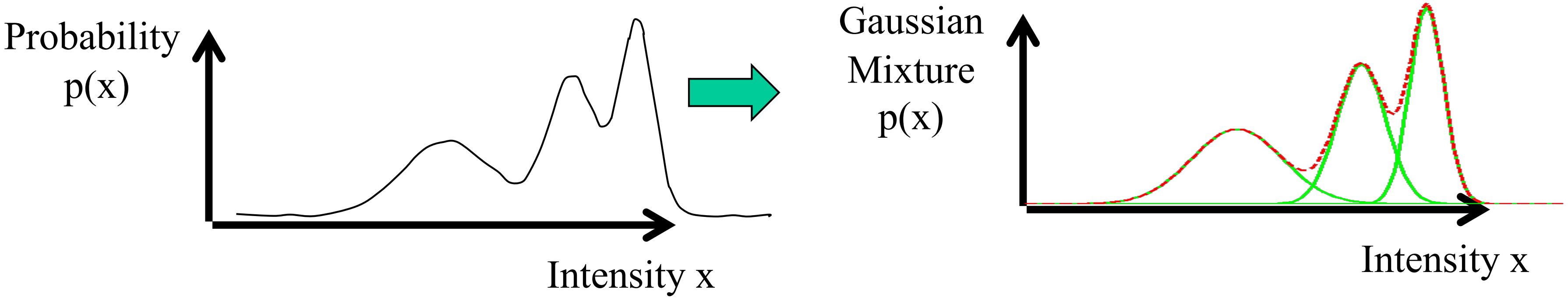
Histogram



Estimation of pdf on x
After kernel-based density
Estimation (Parzen windowing)

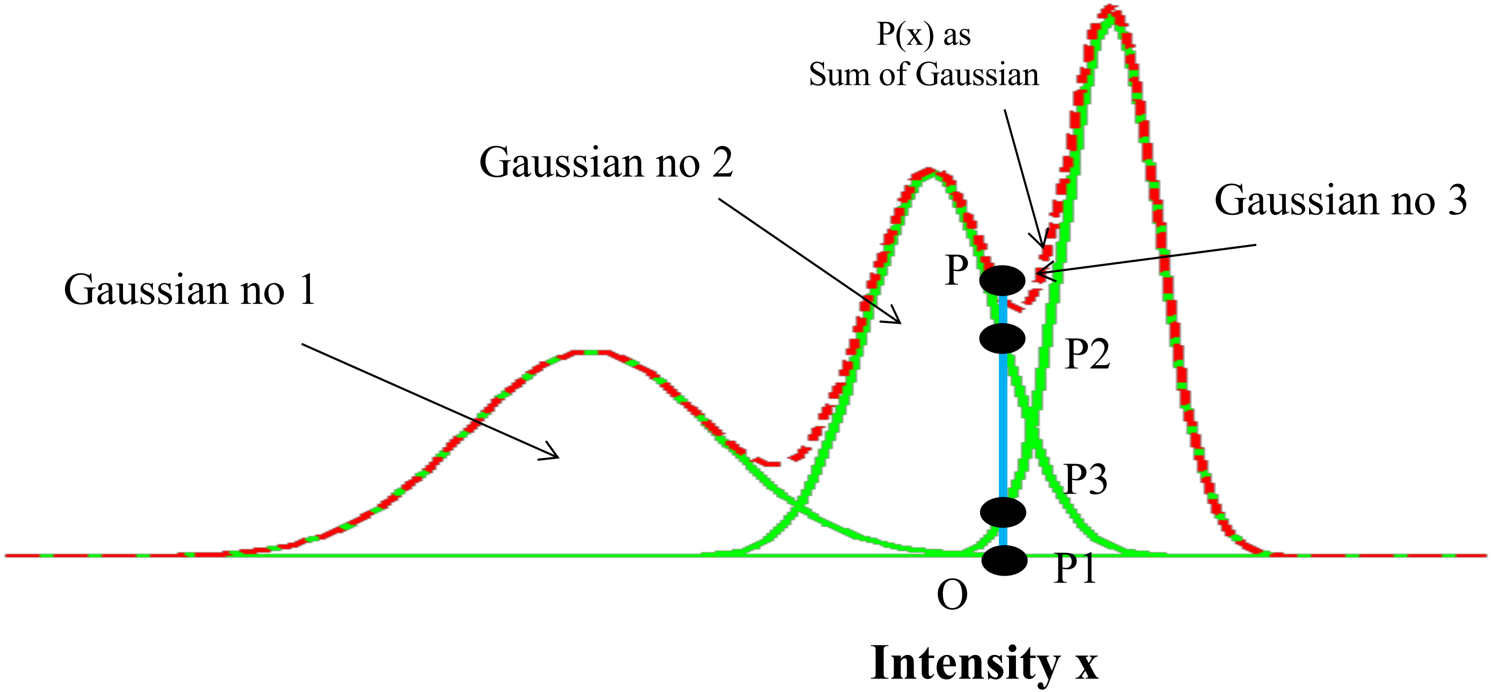
Gaussian Mixture & Histogram

- Assume that the probability p(x) can be decomposed into a sum of Gaussian distributions

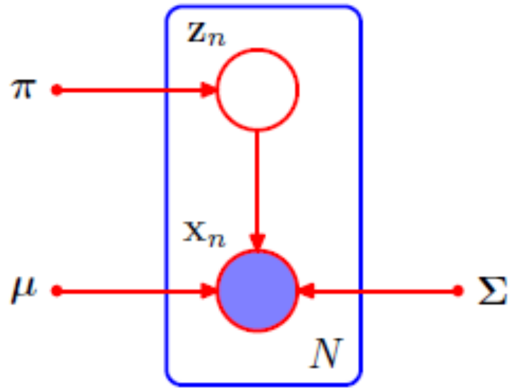


Gaussian Mixture & Histogram

- Interpretation of posterior probability distributions $p(z_k = 1|x) = \frac{OP_k}{OP}$



Problem to solve



- Given
  - Image **X**={x_i} i=1..N
  - Number of classes K
- What are :
  - Gaussian distribution parameters of each class $\theta_I = \{\theta_k\} = \{\mu_k, \Sigma_k\}$
  - Mixture probabilities $\{\pi_k\}$
  - Posterior probabilities

$$p(z_{nk} = 1 | x_n, \theta) \quad \theta = \{\theta_k, \pi_k\}$$

Marginal Likelihood Function

Define the marginal likelihood as the probability of having the data, knowing the parameters

$$\Lambda(\pi,\theta) = p(X \mid \theta) = \prod_{n=1}^{N} p(x_n \mid \theta)$$

Or the (marginal) Log-likelihood L

$$L(\pi,\theta) = \log \Lambda(\pi,\theta) = \sum_n \log(\sum_k \pi_k \mathcal{N}(x_n; \mu_k, \sigma_k))$$

Maximization of Log Likelihood ?

- Classical approach :
  - Write log-Marginal Likelihood of data as a function of θ
  - Coordinate ascent : optimize with respect to each parameter θ_i successively
- Differentiating the log likelihood with μ_k

and set equal to zero gives

$$\frac{\partial \Lambda}{\partial \mu_k} = \sum_{n=1}^{N} \frac{\pi_k G(x_n; \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j G(x_n; \mu_j, \Sigma_j)} \Sigma_k^{-1}(x_n - \mu_k) = 0$$

Giving a non linear function of the unknown parameters.

Cannot be solved in closed form.

3. Medical Image Segmentation

- 3.1 Taxonomy of segmentation algorithms
- 3.2 Validation of segmentation algorithms
- 3.3 Deterministic Filtering & Thresholding Approaches
- 3.4 Probabilistic Imaging Model
- 3.5 **Expectation Maximisation for GMM**
- 3.6 Image classification with bias field
- 3.7 Variational Bayes EM
- 3.8 STAPLE Algorithm

Expectation Maximisation Algorithm

- Iterative approach for estimating parameters of (Gaussian) Mixture parameters
- General Idea :
    - New criterion : Add unknown variable u (posterior) and add constraint (KL divergence)
    - Alternate maximization performed in closed form : equivalent to lower bound maximization

Alternate maximisation

- Replace Log-Likelihood with a criterion easier to optimize but with additional unknowns
- Log-(marginal) likelihood :

$$L(\theta) = \log \Lambda(\theta) = \sum_n \log p(x_n|\theta) = \sum_n \log(\sum_k \pi_k \mathcal{N}(x_n; \mu_k, \sigma_k))$$

- New criterion F(θ, u):
  - Add $u = \{u_{nk}\}$ as unknown. u is a vector of u_{nk} which is the posterior probability

$$\mathrm{F}(\theta, u) = \mathrm{L}(\theta) - D_{KL}\big(u||p(z|x)\big)$$

- By maximizing F with respect to u,

$$u_{nk} = p(z_{nk} = 1|x_n)$$

Why is it easier to optimize $\mathrm{F}(\theta, u)$?

- General result :
  - X = observed random variable
  - Z = hidden random variable
  - Joint probability $p(x_n, z_n) = p(x_n|z_n)p(z_n) = p(z_n|x_n)p(x_n)$
  - Constraint on u_{nk}: $\sum_k u_{nk} = 1$
  - Log likelihood : $L(\theta) = \sum_n \log p(x_n) = \sum_n \sum_k u_{nk} \log p(x_n)$
  - New criterion :

$$\mathrm{F}(\theta, u) = \sum_n \sum_k u_{nk} \log p(x_n) - \sum_n \sum_k u_{nk} \log u_{nk}/p(z_{nk}|x_n)$$

$$\boxed{\mathrm{F}(\theta, u) = \sum_n \sum_k u_{nk} \log p(x_n, z_{nk}) - \sum_n \sum_k u_{nk} \log u_{nk}}$$

We have « relaxed » the optimization problem by introducing
a new unknown variable

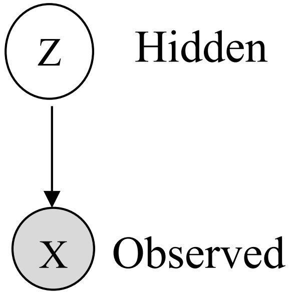
Interpretation

- New criterion involves 2 terms :

$$\mathrm{F}(\theta,u) = \underbrace{\textstyle\sum_n \sum_k u_{nk} \log p(x_{nk}, z_{nk})}_{Q(\theta,u)} - \underbrace{\textstyle\sum_n \sum_k u_{nk} \log u_{nk}}_{\mathbb{H}(u)}$$

  - F(θ, u) is the *variational lower bound*
  - -F(θ, u) is the *variational free energy*= average energy - entropy
  - $Q(\theta, u) = \sum_n \sum_k u_{nk} \log p(x_{nk}, z_{nk}) = \mathbb{E}_U(\log p(X, Z))$ is the expectation of the complete likelihood
  - $\mathbb{H}(u) = -\sum_n \sum_k u_{nk} \log u_{nk}$ is the **entropy** of the approximate posterior probability
- $Q(\theta, u)$ is easier to optimize wrt θ because it involves complete likelihood = likelihood of observed and hidden variables

Evidence Lower Bound



- General result :
  - For any inverse problem where Z is the hidden variable and X observed variable :

$$\boxed{\begin{aligned} &\log p(X) - D_{KL}\big(u \| p(Z|X)\big) \\ &= \mathbb{E}_u(\log p(X,Z)) + \mathbb{H}(u) \end{aligned}}$$

- Variational lower bound :

$$\log \mathrm{p}(X) \geq \mathbb{E}_u(\log p(X,Z)) + \mathbb{H}(u)$$

Case of Gaussian Mixtures

- Log likelihood

  $L(\theta) = \log \Lambda(\theta) = \sum_n \log(\sum_k \pi_k \mathcal{N}(x_n; \mu_k, \sigma_k))$

- Function of parameters :

  $$Q(\theta, u) = \sum_n \sum_k u_{nk} \log \pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)$$

- Note that we have sum of log instead of log of sums !
- Criterion F$(\theta, u) = \mathrm{Q}(\theta, u) + \mathbb{H}(u)$is known as **Hathaway criterion**

EM Algorithm

- The algorithm optimizes alternatively between u and θ = coordinate ascent

$$\mathrm{F}(\theta, u) = \mathrm{L}(\theta) - D_{KL}\big(u||p(z|x)\big) = \mathrm{Q}(\theta, u) + \mathbb{H}(u)$$

- Constraints : $\sum_k \pi_k = 1 \quad \sum_k u_{nk} = 1$
- E-step
  - maximize $F(\theta, u)$ wrt u

$$\text{Compute } \boxed{\mathrm{u_{nk}} = \frac{\pi_k \mathcal{N}(x_n|\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \pi_k\, \mathcal{N}(x_n|\mu_k,\Sigma_k)}}$$

  - Equivalent to minimizing KL divergence between u and posterior probability

M-Step

- M-step : maximize $F(\theta, u)$ or equivalently $Q(\theta, u)$ wrt $\theta = \{\theta_S, \theta_I\}$
  - Optimize with respect to mean μ_k

$$\frac{\partial Q}{\partial \mu_k} = 0 \quad \Rightarrow \quad \boxed{\mu_k = \frac{\sum_{n=1}^{N} u_{nk} \, x_n}{\sum_{n=1}^{N} u_{nk}}}$$

  - Optimize with respect to covariance Σ_k

$$\frac{\partial Q}{\partial \Sigma_k} = 0 \quad \Rightarrow \quad \boxed{\Sigma_k = \frac{\sum_{n=1}^{N} u_{nk} (x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{i=1}^{N} u_{nk}}}$$

  - Optimize with respect to prior probabilities

$$\frac{\partial Q}{\partial \pi_k} = 0 \quad \Rightarrow \quad \boxed{\pi_k = \frac{1}{N} \sum_{n=1}^{N} u_{nk}}$$

EM Algorithm for GMM

- Iterative scheme
  - Make initial guesses for the parameters
  - Alternate between the following two stages:
    1. E-step: evaluate posterior u_{nk}

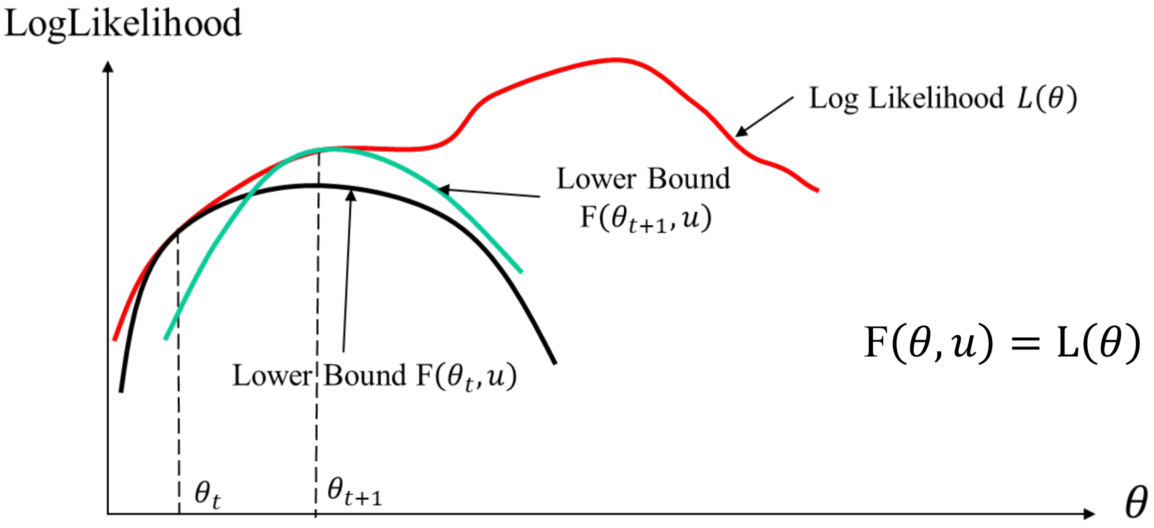
$$\mathrm{u}_{\mathrm{nk}} = \frac{\pi_k \mathcal{N}(x_n|\mu_k,\Sigma_k)}{\sum_{k=1}^{K} \pi_k \, \mathcal{N}(x_n|\mu_k,\Sigma_k)}$$

    2. M-step: update parameters (μ_k,Σ_k,π_k) using ML results

$$\mu_k = \frac{\sum_{n=1}^{N} u_{nk} \; x_n}{\sum_{n=1}^{N} u_{nk}} \qquad \Sigma_k = \frac{\sum_{n=1}^{N} u_{nk} \, (x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{i=1}^{N} u_{nk}} \qquad \pi_k = \frac{1}{N} \sum_{n=1}^{N} u_{nk}$$

EM as Iterated Lower Bound Maximisation

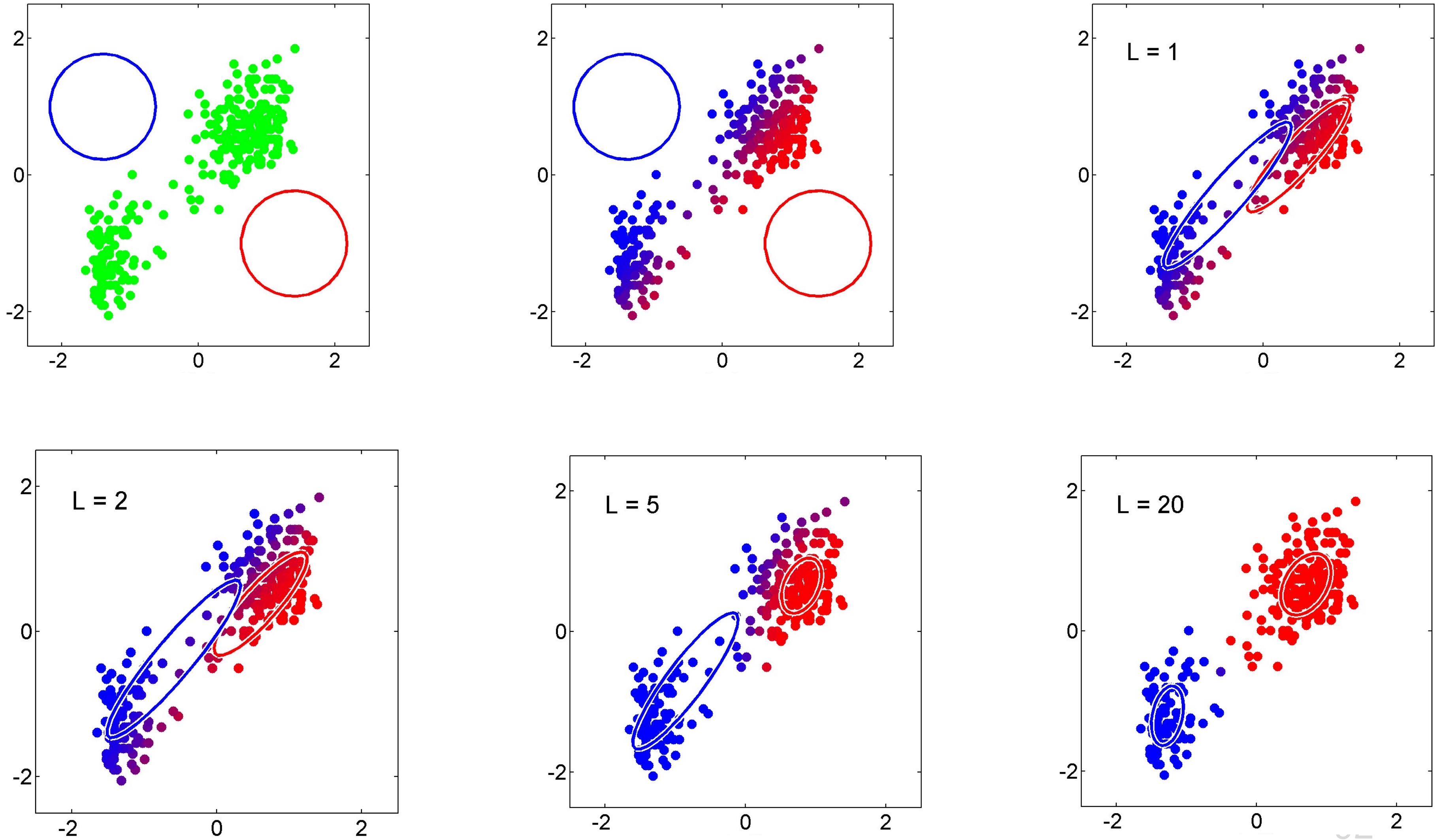
- Equivalent view of EM algorithm :
  - E-step leads to $u = p(z|x)$ and therefore makes $L(\theta_t) = F(\theta_t, u)$.
  - $F(\theta, u)$ is a lower bound of Log-likelihood $L(\theta)$ since Kullback Leibler divergence is positive
  - M-step optimizes $F(\theta, u)$ with respect to θ which is easier to maximize than log likelihood



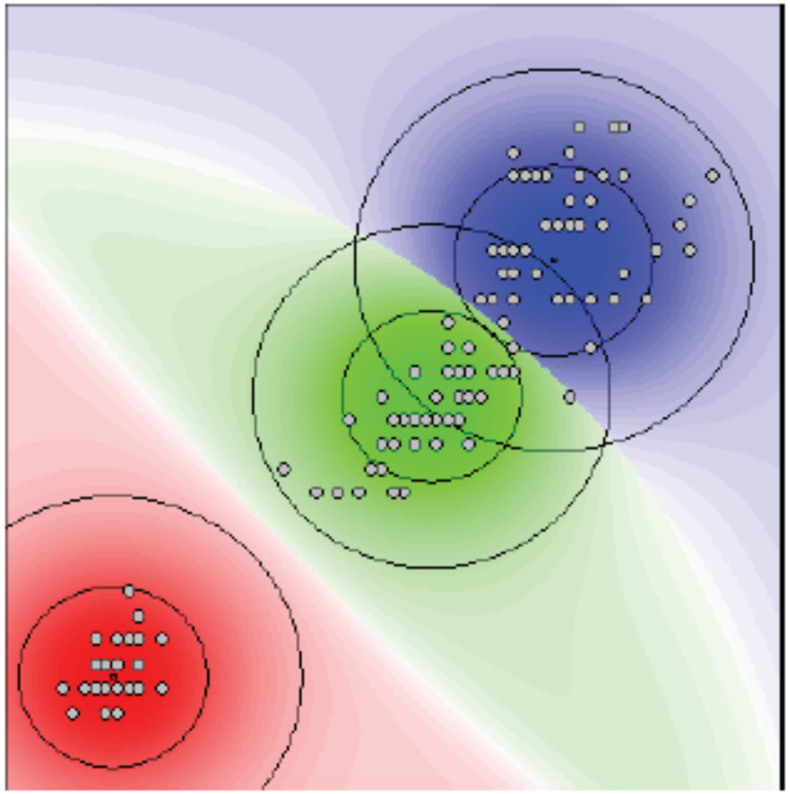
$$\mathrm{L}(\theta) \geq \mathrm{F}(\theta, u)$$

$$\mathrm{F}(\theta, u) = \mathrm{L}(\theta) - D_{KL}\big(u||p(z|x)\big) = \mathrm{Q}(\theta, u) + \mathbb{H}(u)$$

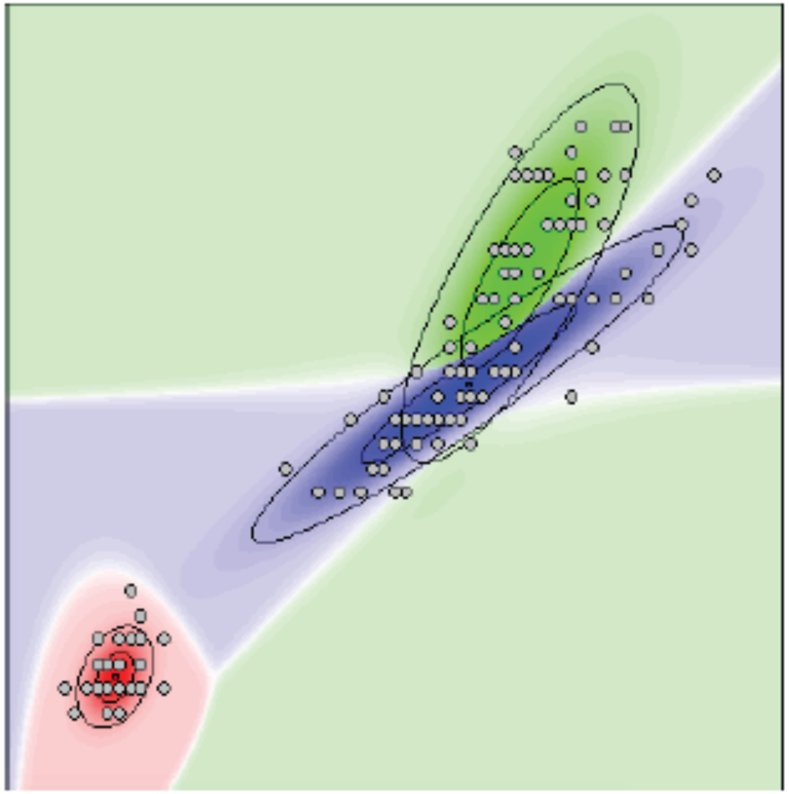
Example of EM with 2 Gaussian distributions



EM on Iris data



equal prior, spherical

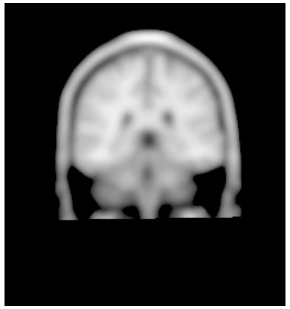


equal prior, ellipsoidal

Class Priors

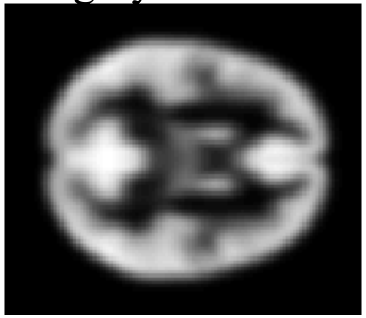
- Initial hypothesis : homogeneous priors $p(z_{nk} = 1) = \pi_k$ is estimated
- Priors may be given by atlas registered on images. In this case θ_s are the registration parameters

Atlas



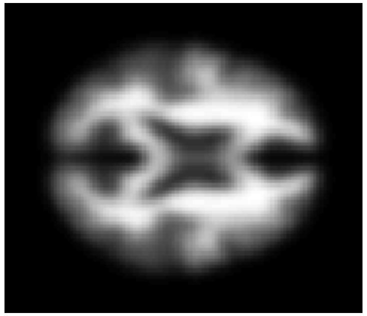
T1 template

Prior $p(z_{n1})$ on grey matter



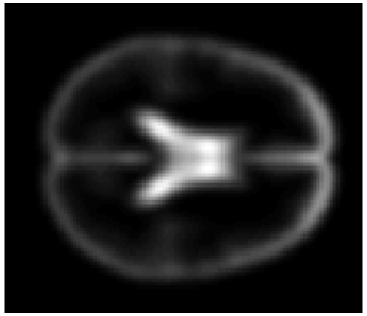
gray matter

Prior $p(z_{n2})$ on White matter



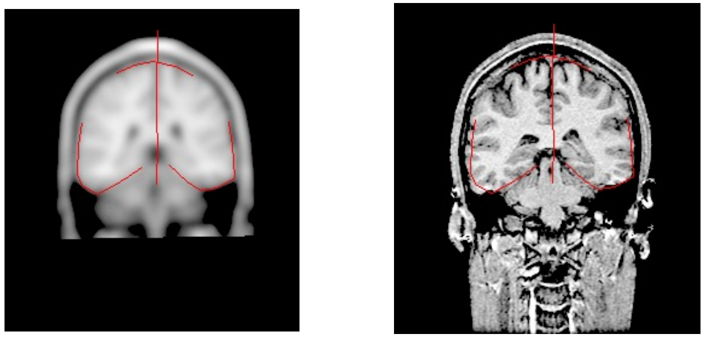
white matter

Prior $p(z_{n3})$ on cerebro spinal fluid



csf

Affinely Registered Atlas



Courtesy of D. Vandermeulen

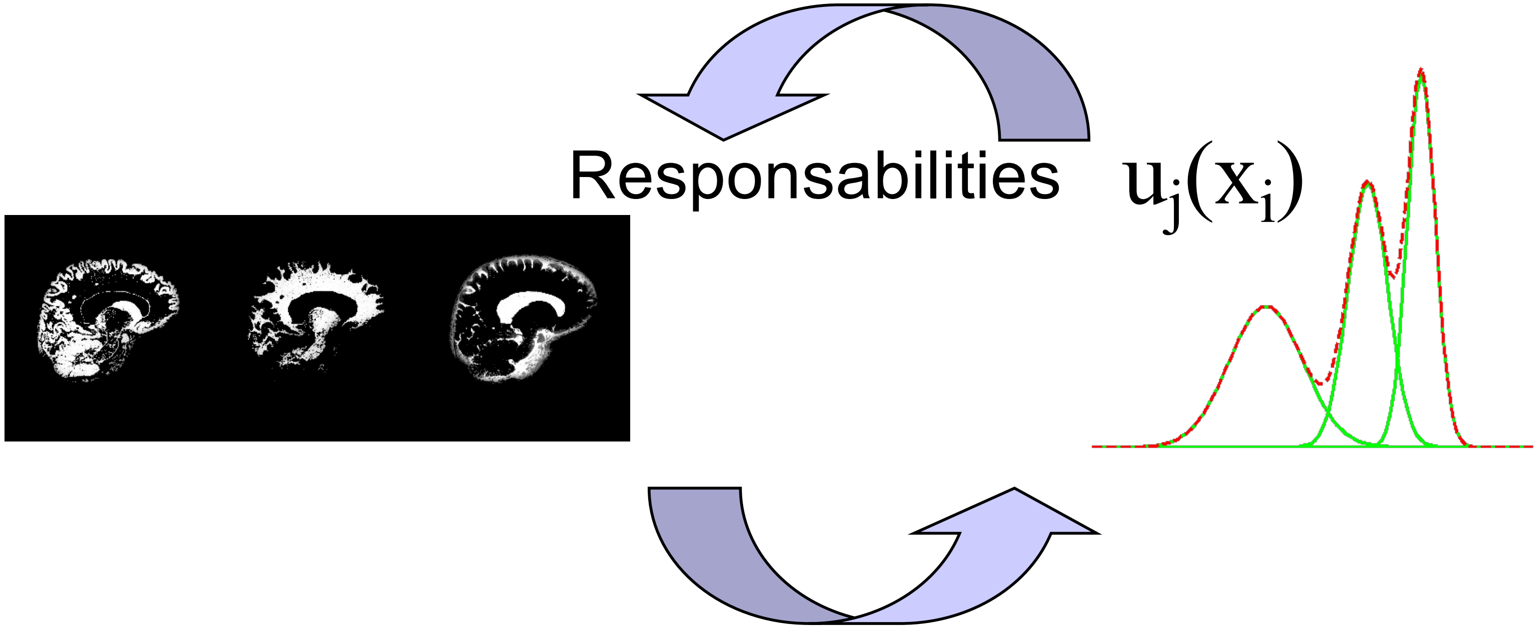
Example : BrainWeb at MNi

http://www.bic.mni.mcgill.ca/brainweb/

EM for Image Intensity Classification

- Use the EM algorithm [Dempster77,Wells94] :

Expectation-Maximisation

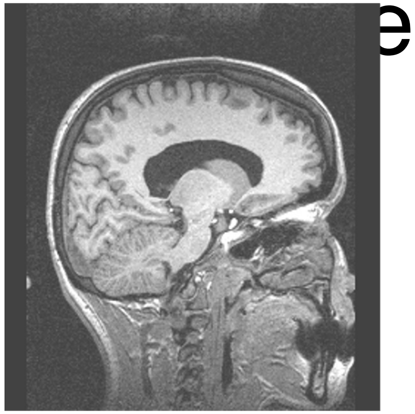


Responsabilities $u_j(x_i)$

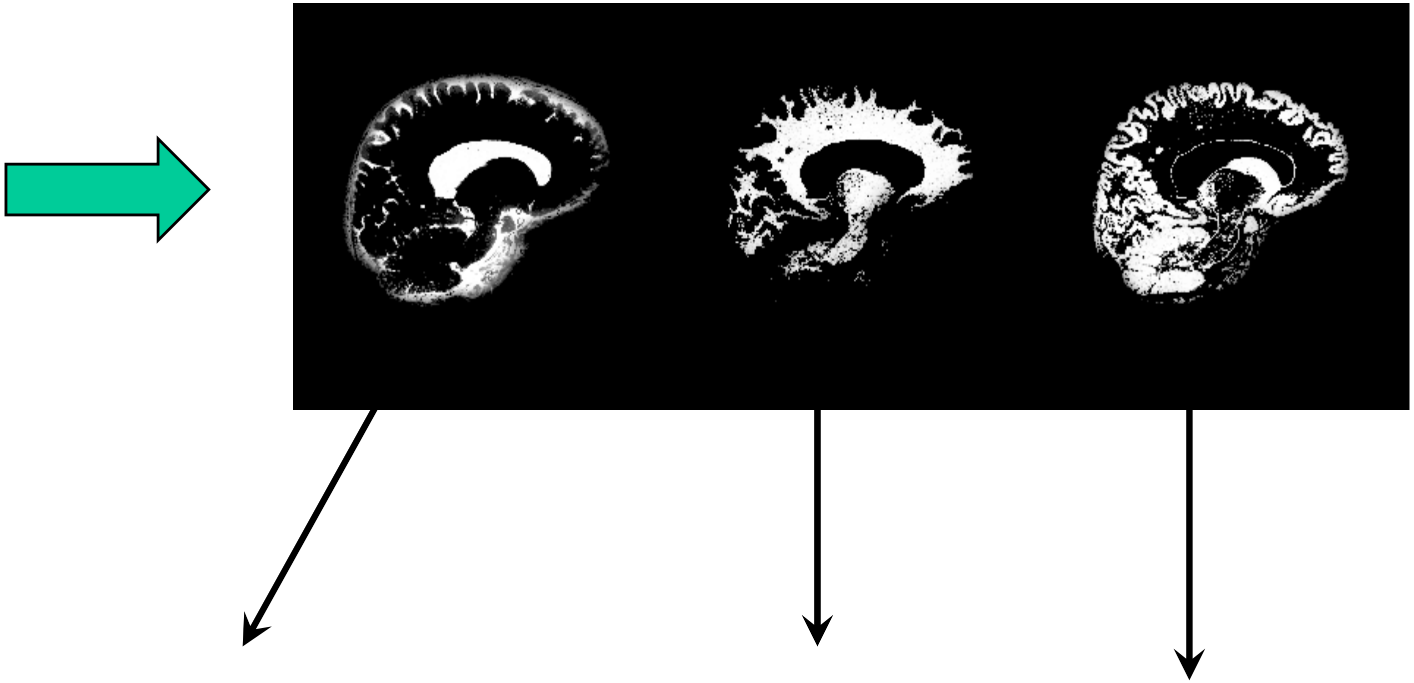
Mixture Param. μ_κ Σ_κ π_κ

Brain Tissue Classification

- Typical application : use MR cerebral image



Courtesy of D. Vandermeulen

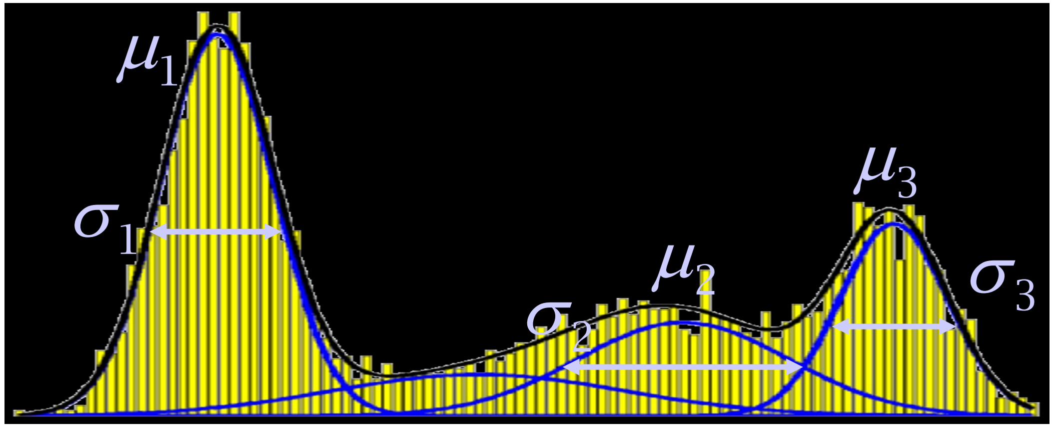


Cerebro-spinal fluid Grey matter White matter

3 Classes

Scalar feature
= Intensity

EM Classification - Algorithm



Starting estimates for responsibilities

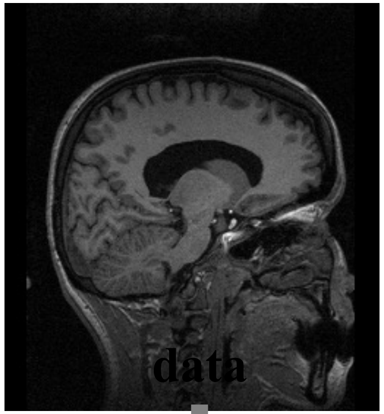
Compute Gaussian parameters from Responsibilities

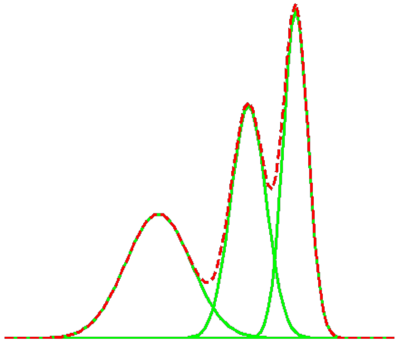
Compute Responsibilities from Gaussian parameters

Converged ?
No
Yes

STOP

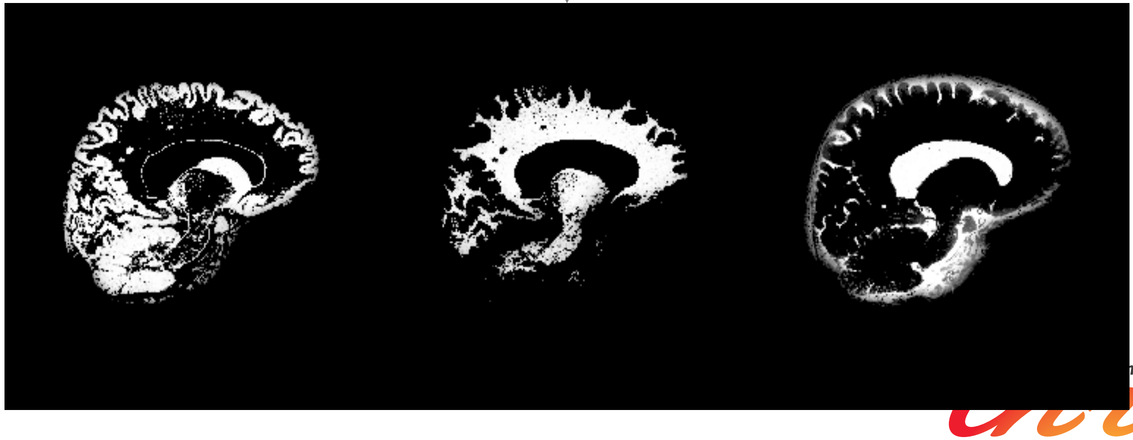
Stage 1: Expectation


data


distribution

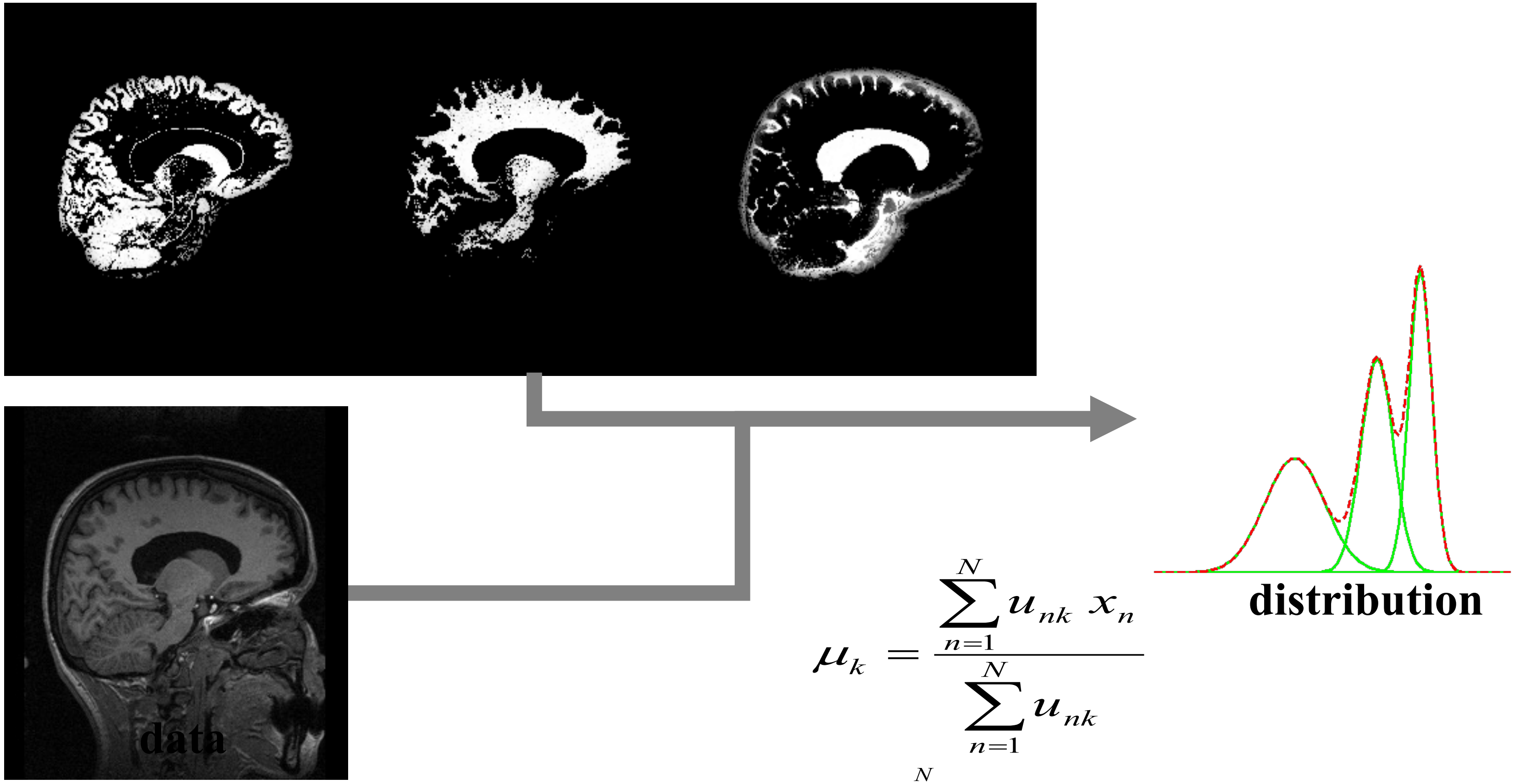
Compute
Responsibilities

$$\mathrm{u_{nk}} = \frac{\pi_k \mathcal{N}(x_n | \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \, \mathcal{N}(x_n | \mu_k, \Sigma_k)}$$



Courtesy of D. Vandermeulen

Stage 2: Maximization



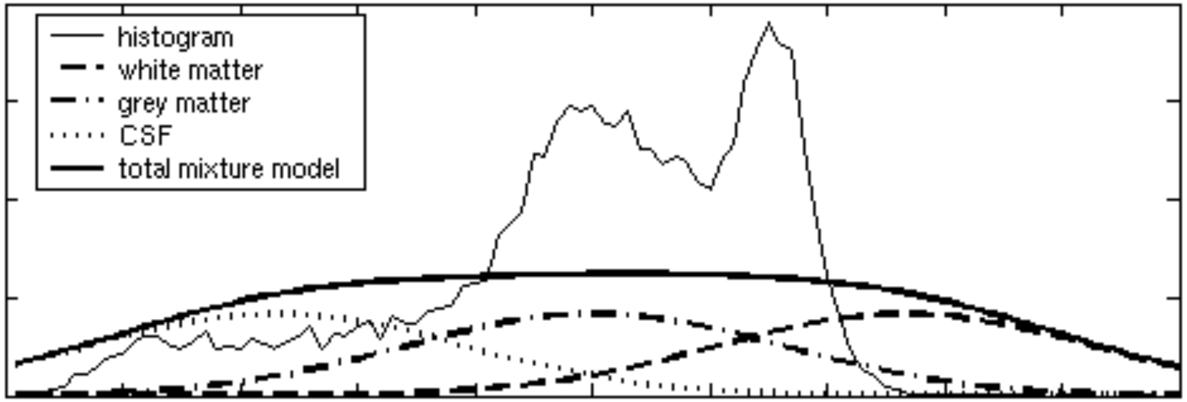
$$\mu_k = \frac{\sum_{n=1}^{N} u_{nk} \; x_n}{\sum_{n=1}^{N} u_{nk}}$$

$$\Sigma_k = \frac{\sum_{n=1}^{N} u_{nk} (x_n - \mu_k)(x_n - \mu_k)^T}{\sum_{i=1}^{N} u_{nk}}$$

$$\pi_k = \frac{1}{N} \sum_{n=1}^{N} u_{nk}$$

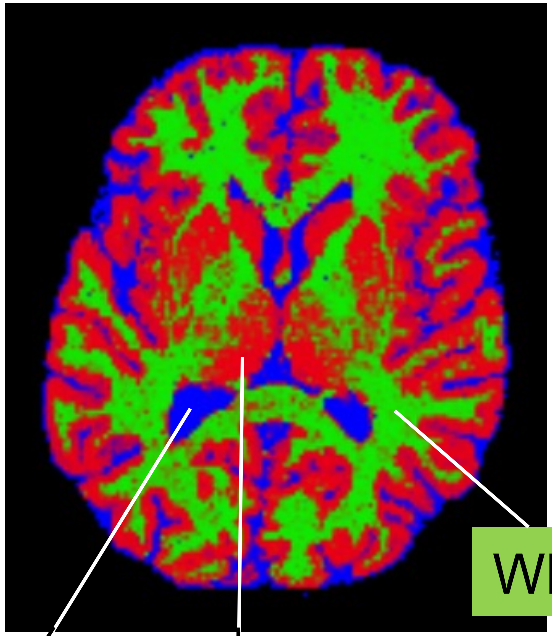
Courtesy of D. Vandermeulen

Iterations EM



Courtesy of K. Van Leemput

Results

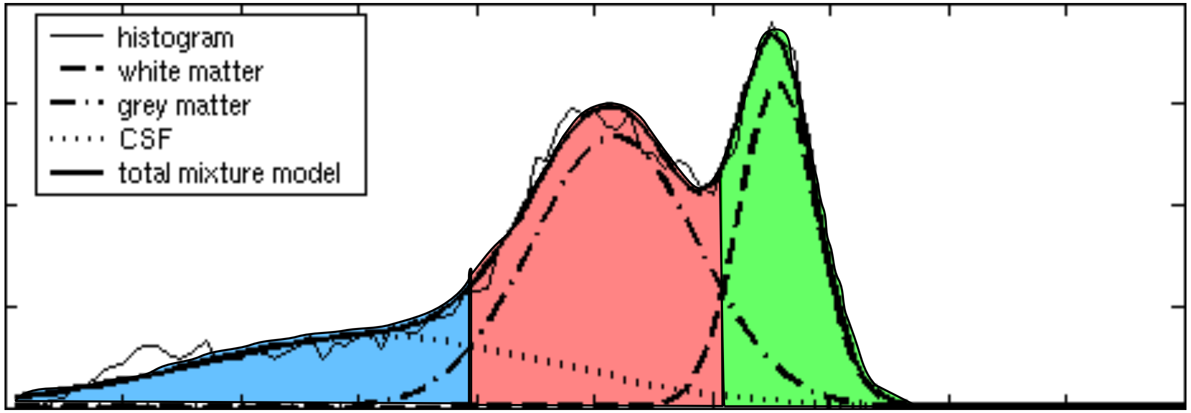


White matter

CSF

Grey Matter

Courtesy of K. Van Leemput



$$\hat{\mathbf{l}} = \arg\max_{\mathbf{l}} p(\mathbf{l}|\mathbf{d}, \hat{\boldsymbol{\theta}})$$

GMM and K-Means

- GMM with :
  - Isotropic variance $\Sigma_k = \epsilon \, Id$
  - Uniform prior : $\pi_k = \frac{1}{K}$
- Expectation of complete Lik. : $Q(\theta) = -\sum_n \sum_k \frac{u_{nk}|x_n - \mu_k|^2}{2\epsilon}$
- Same as Fuzzy-Cmeans with m=1
- Same as K-means when :
  - $\epsilon \rightarrow 0$
  - $u_{nk} \in \{0,1\}$

$$\mathrm{u_{nk}} = \frac{\exp\left(-|x_n - \mu_k||^2/2\epsilon\right)}{\sum_{j=1}^{K} \exp\left(-||x_n - \mu_j||^2/2\epsilon\right)} \rightarrow r_{nk} \in \{0,1\}$$

K Means functional

- K Means algorithm consists in optimizing the functional :
  - $\mathrm{J}(r, \mu) = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \, \|x_n - \mu_k\|^2$
  - With the constraint that $r_{nk} \in \{0,1\}$ and $\sum_{k=1}^{K} r_{nk} = 1 \ \forall n$
- J can be seen as
  - minimizing the correlation between the assignment and the distance to cluster center
  - Minimizing the compactness of the clusters

K Means optimization

- Perform alternate optimization :
  - Consider μ_k fixed and optimize on r_{nk}
    - For each data x_n choose which r_{nk} is 1

E-Step

$$r_{nk} = \begin{cases} 1 \text{ if } \mathrm{k} = \arg min_j \|x_n - \mu_k\| \\ 0 \text{ otherwise} \end{cases}$$

  - Consider r_{nk} fixed and optimize on μ_k

M-Step

$$\frac{\partial J}{\partial \mu_k} = 2 \sum_{n=1}^{N} r_{nk}(\mu_k - x_n) = 0$$

$$\mu_k = \frac{\sum_{n=1}^{N} r_{nk} x_n}{\sum_{n=1}^{N} r_{nk}}$$

Good Initial Seeds (kmeans++)

- Choose the centers as far away as possible from each other but in a random manner.
- Algorithm :
  - Choose one center at random μ_1
  - While $k \leq K$
    - Compute $\mathrm{d_n} = \arg min_{j<k} \left\|x_n - \mu_j\right\|^2$ the minimum distance of data x_n to the already chosen centers
    - Pick μ_k among data with probability proportional to d_n
    - k=k++

David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. “Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms“, 2007 , pp. 1027–1035

Issues with EM for GMM

- Presence of bias field in MR images
- EM leads to only local maxima of Log-likelihood
- Functional admits trivial solutions (zero covariance centered at data points) that can lead to bad estimate
- The covariance matrix Σ_k should be invertible which is not guaranteed (may use pseudo-inverse)
- How to choose the number of classes
- How to make the estimation robust
  to outliers ?