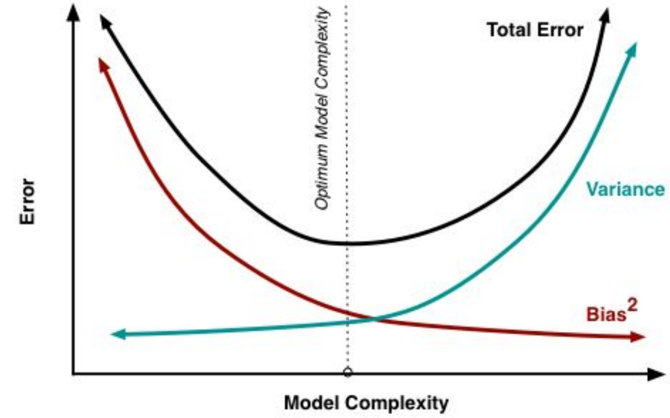
GMM & EM

Last time summary

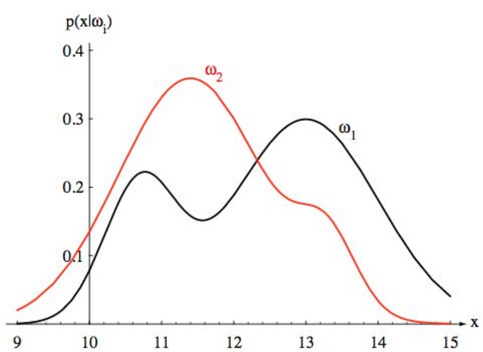
- Normalization
- Bias-Variance trade-off
  - Overfitting and underfitting
- MLE vs MAP estimate
  - How to use the prior
- LRT (Bayes Classifier)
  - Naïve Bayes

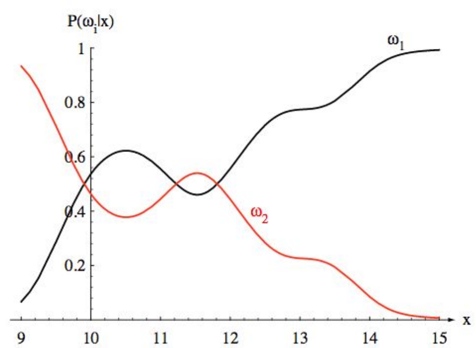


$$\boxed{\frac{P(x|w_1)}{P(x|w_2)}} \quad ? \quad \boxed{\frac{P(w_2)}{P(w_1)}}$$

Likelihood ratio

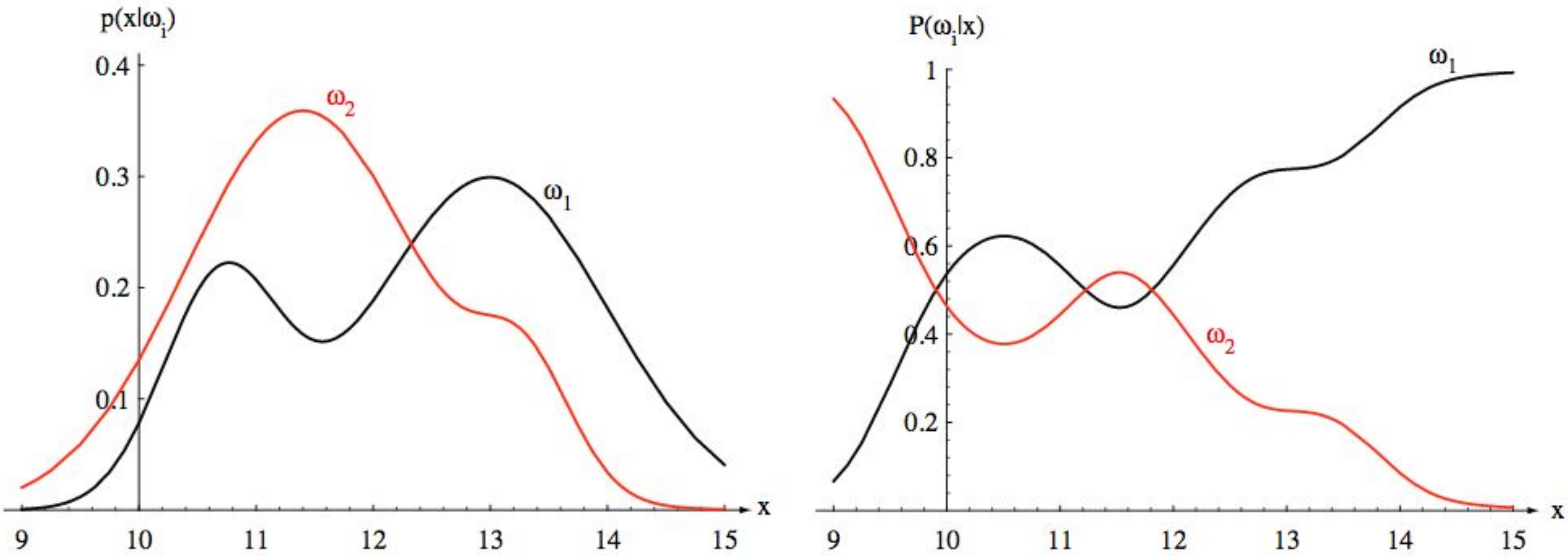
Ratio of priors





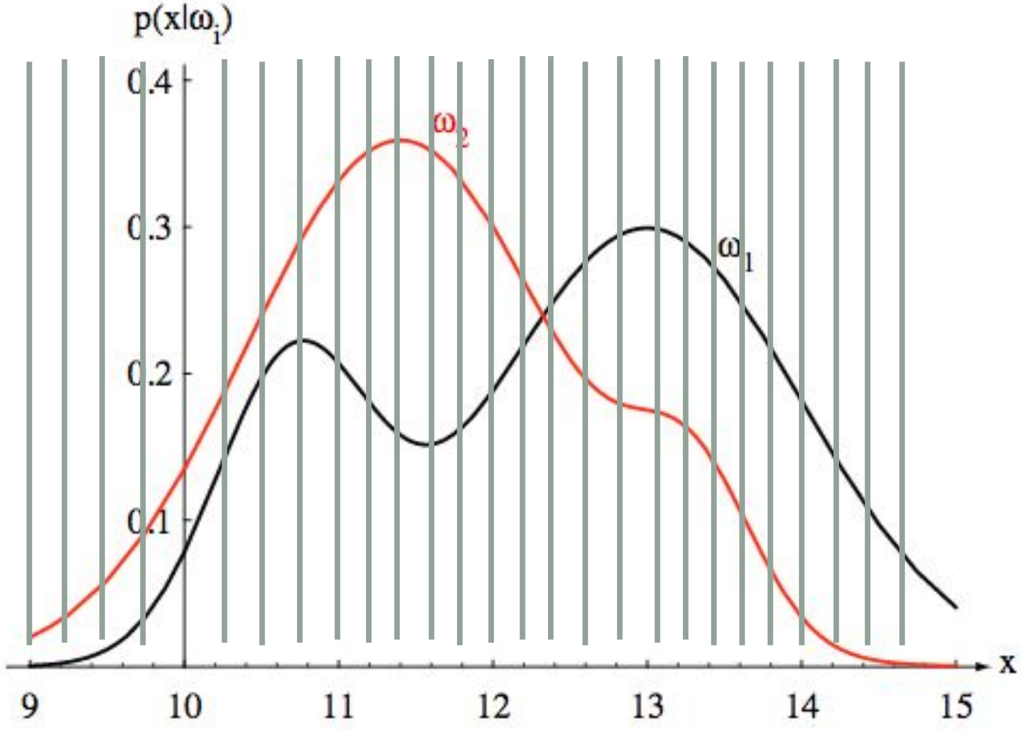
A simple decision rule

- If we can know either p(x|w) or p(w|x) we can make a classification guess



Goal: Find p(x|w) or p(w|x) by finding the parameter of the distribution

A simple way to estimate p(x|w)



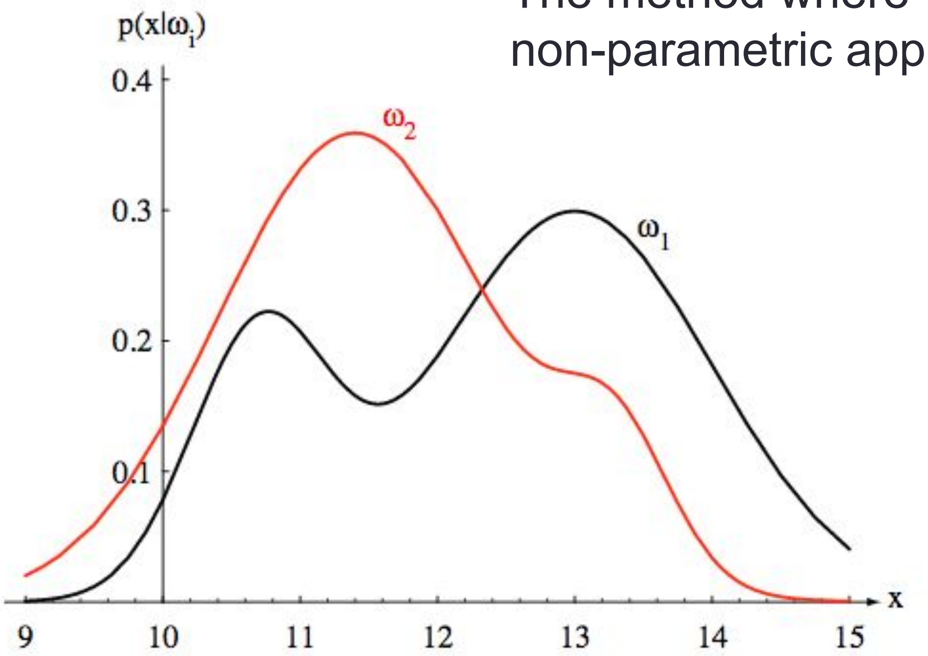
Make a histogram!

What happens if there is no data in a bin?

The parametric approach

- We assume p(x|w) or p(w|x) follow some distributions with parameter θ

The method where we find the histogram is the non-parametric approach



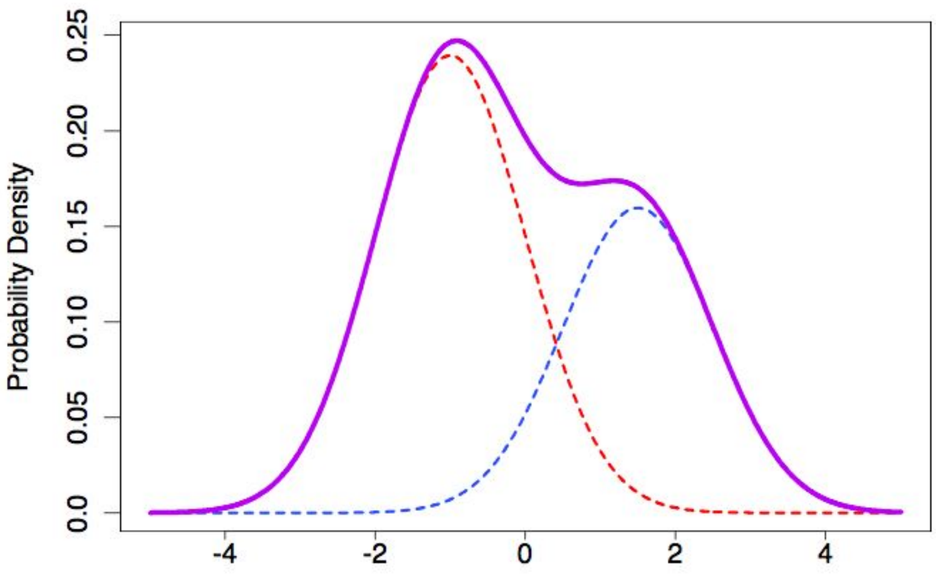
$$\boxed{p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2}}$$

Goal: Find θ so that we can estimate p(x|w) or p(w|x)

Gaussian Mixture Models (GMMs)

- Gaussians cannot handle multi-modal data well
- Consider a class can be further divided into additional factors
- Mixing weight makes sure the overall probability sums to 1

$$P(x) \sim \sum_{k=1}^{K} w_k N(\mu_k, \sigma_k)$$

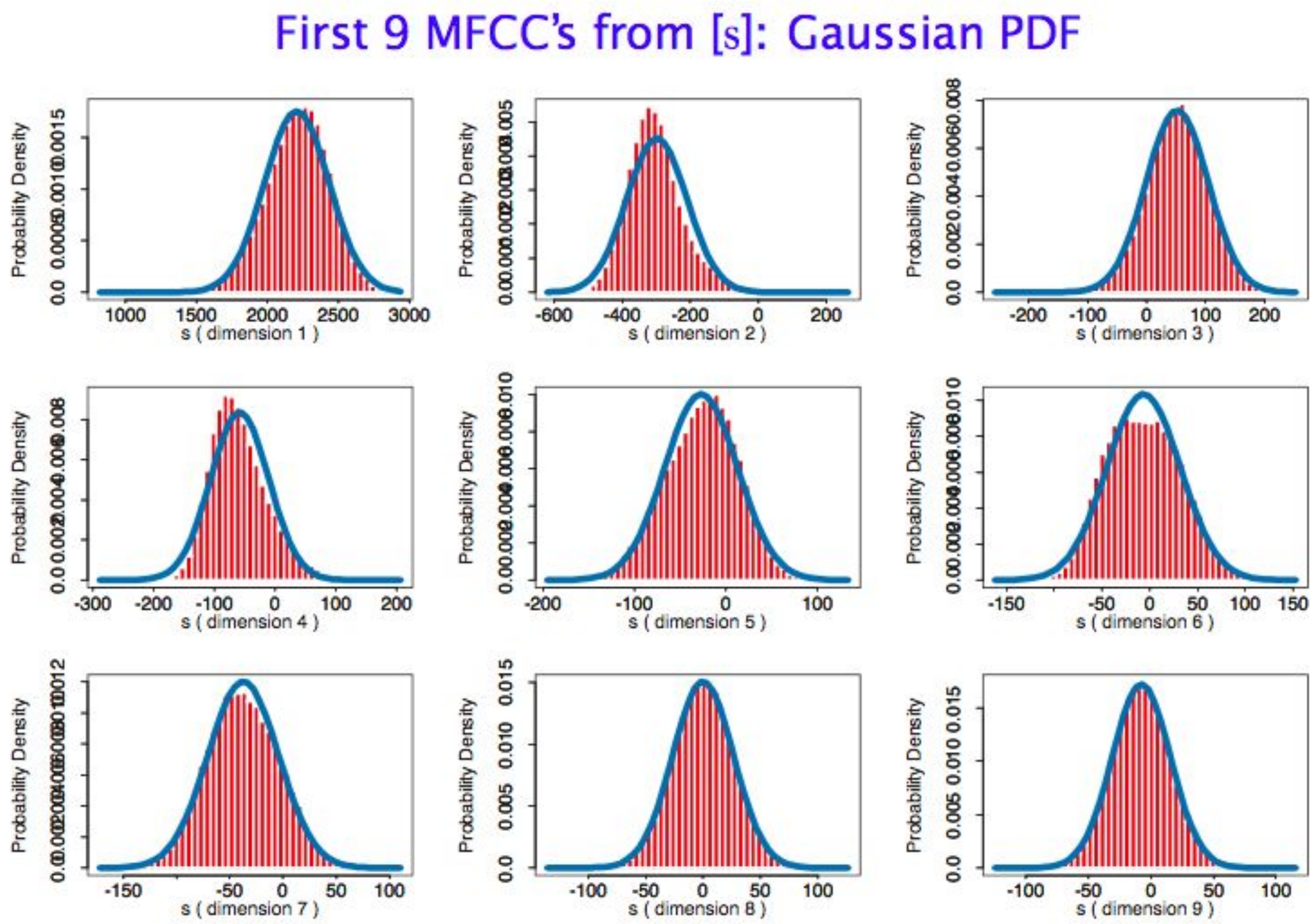


$\frac{x}{\sigma}$

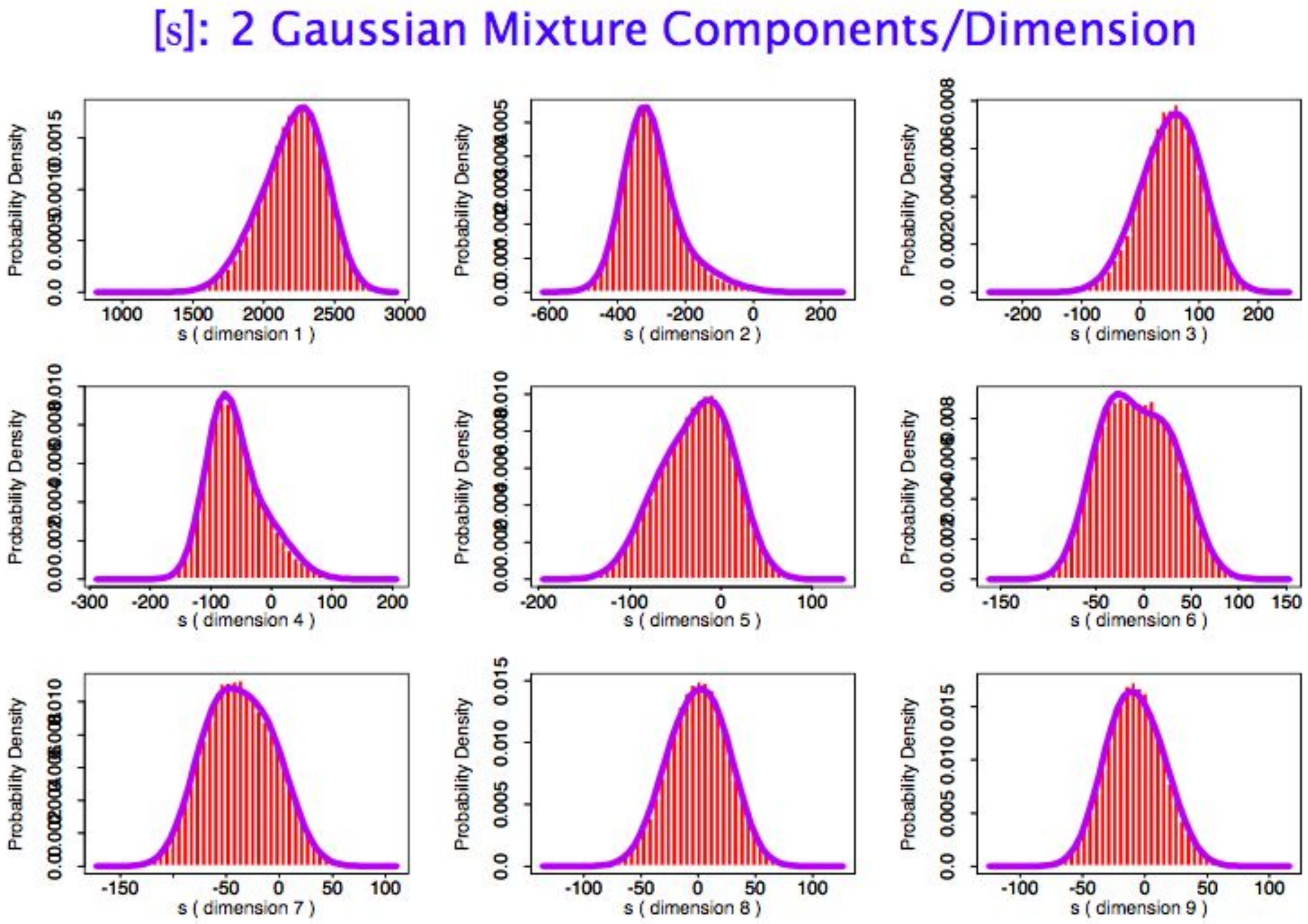
$$p(x) = 0.6 p_1(x) + 0.4 p_2(x)$$

$$p_1(x) \sim N(-\sigma, \sigma^2) \qquad p_2(x) \sim N(1.5\sigma, \sigma^2)$$

Model of one Gaussian



Mixture of two Gaussians

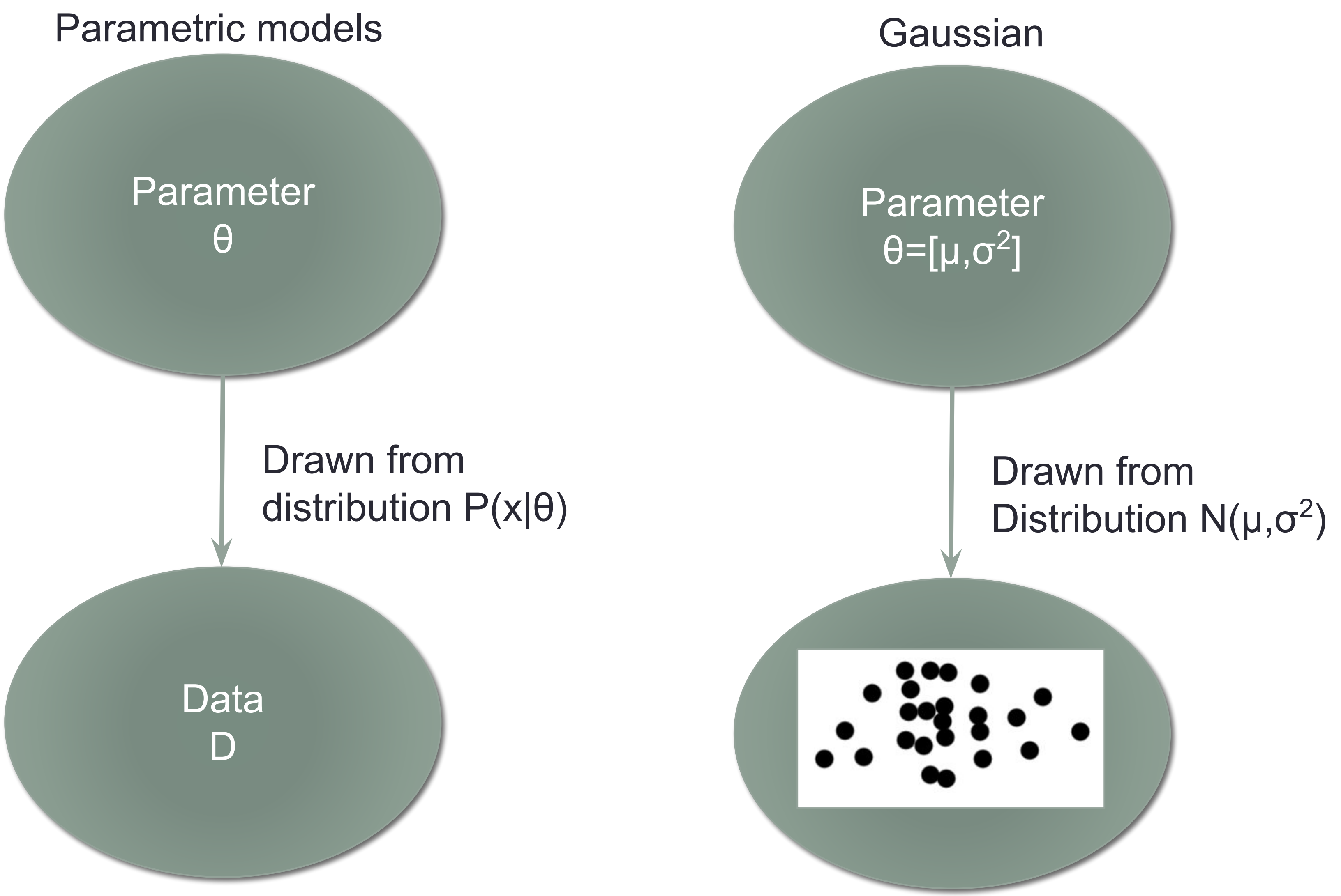


Mixture models

$$p(x) = \sum_k p(k) p_k(x)$$

- A mixture of models from the same distributions (but with different parameters)
- Different mixtures can come from different sub-class
  - Cat class
    - Siamese cats
    - Persian cats
- p(k) is usually categorical (discrete classes)
- Usually the exact class for a sample point is unknown.
  - Latent variable

Parametric models



Maximum A Posteriori (MAP) Estimate

MLE

- Maximizing the likelihood (probability of data given model parameters)

$\underset{\theta}{\text{argmax}}\ p(\mathbf{x}|\theta)$

$p(\mathbf{x}|\theta)$
$= L(\theta)$

- Usually done on log likelihood

- Take the partial derivative wrt to θ and solve for the θ that maximizes the likelihood

MAP

- Maximizing the posterior (model parameters given data)

$\underset{\theta}{\text{argmax}}\ p(\theta|\mathbf{x})$

- But we don't know $p(\theta|\mathbf{x})$

- Use Bayes rule

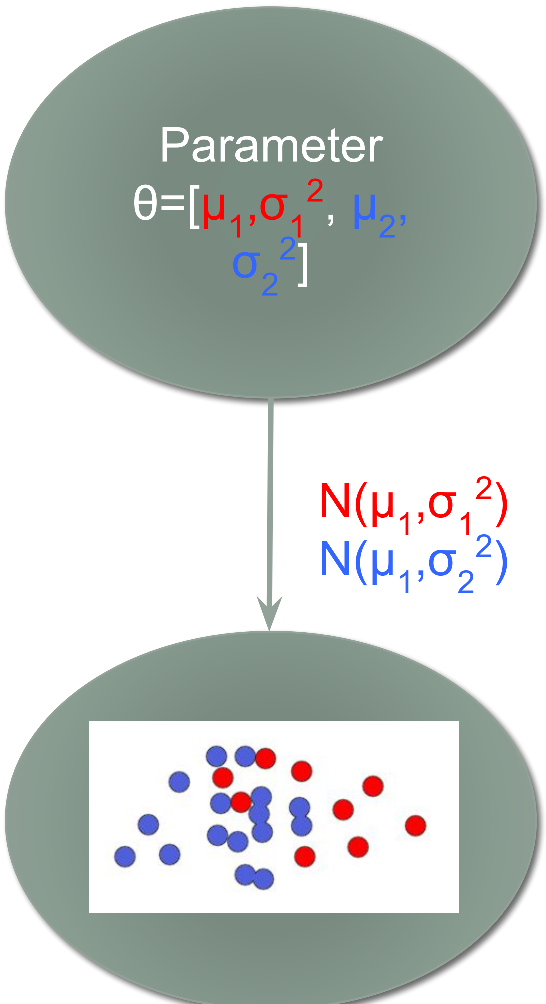
$$p(\theta|\mathbf{x}) = \frac{p(\mathbf{x}|\theta)p(\theta)}{p(\mathbf{x})}$$

- Taking the argmax for θ we can ignore $p(\mathbf{x})$

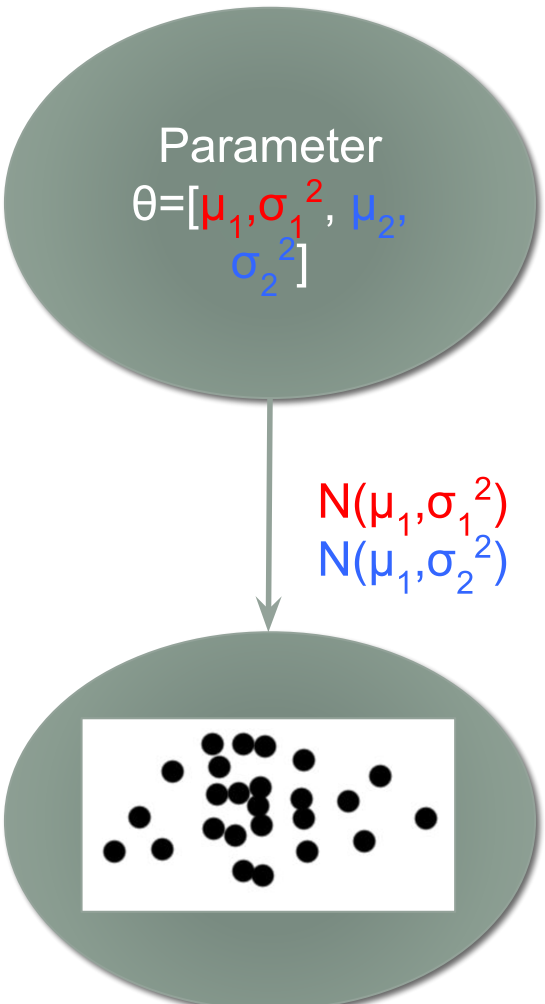
- $\underset{\theta}{\text{argmax}}\ p(\mathbf{x}|\theta)\ p(\theta)$

What if some data is missing?

Mixture of Gaussian

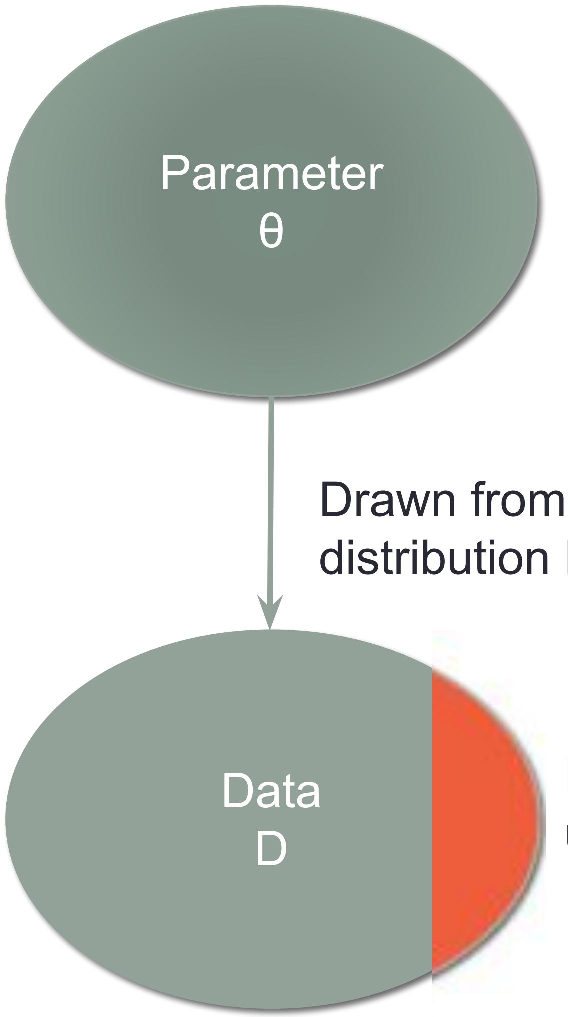


Unknown mixture labels



Estimating missing data

Parametric models



Drawn from distribution $P(x,k|\theta)$

Latent variables,k unknown

Need to estimate both the latent Variables and the model parameters.

Estimating latent variables and model parameters

- GMM $$p(x) = \sum_k p(k) N(\mu_k, \sigma_k)$$
- Observed $(x_1, x_2, \ldots, x_N)$
- Latent $(k_1, k_2, \ldots, k_N)$ from K possible mixtures
- Parameter for p(k) is ϕ , $p(k = 1) = \phi_1$, $p(k = 2) = \phi_2 \ldots$

$$l(\phi, \mu, \Sigma) = \Sigma_{n=1}^{N} logp(x^{(i)}; \phi, \mu, \sigma)$$

$$= \Sigma_{n=1}^{N} log \boxed{\Sigma_{l=1}^{K}} p(x_n | k_{n,l}; \mu, \sigma) p(k_{n,l}; \phi)$$

Make things hard to solve

Cannot be solved by differentiating

Assuming k

- What if we somehow know k_n?
- Maximizing wrt to ϕ, μ, σ gives

$$\phi_j = \frac{1}{N}\Sigma_{n=1}^{N} 1(k_n = j)$$

$$\mu_j = \frac{\Sigma_{n=1}^{N} 1(k_n = j) x_n}{\Sigma_{n=1}^{N} 1(k_n = j)}$$

$$\sigma_j^2 = \frac{\Sigma_{n=1}^{N} 1(k_n = j)(x_n - \mu_j)^2}{\Sigma_{n=1}^{N} 1(k_n = j)}$$

$$1(condition)$$

Indicator function. Equals one if condition is met. Zero otherwise

Iterative algorithm

- Initialize ϕ, μ, σ
- Repeat till convergence
  - Expectation step (E-step) : Estimate the latent labels **k**
  - Maximization step (M-step) : Estimate the parameters ϕ, μ, σ given the latent labels
- Called Expectation Maximization (EM) Algorithm
- How to estimate the latent labels?

Iterative algorithm

- Initialize ϕ, μ, σ
- Repeat till convergence
  - Expectation step (E-step) : Estimate the latent labels **k** by finding the pdf of k given everything else $p(k| \phi, \mu, \sigma, x)$
  - Maximization step (M-step) : Estimate the parameters ϕ, μ, σ given the latent labels by maximizing the expectation of the log likelihood
- Extension of MLE for latent variables
  - MLE : $\text{argmax} \log p(x|\theta)$
  - EM : $\text{argmax}\ E_k[\log p(x, k|\theta)]$

Iterative algorithm (general)

- Goal of EM : argmax log $\sum_k$ p(x, k|θ) ≥ argmax E_k[log p(x, k|θ)]
-
- Initialize ϴ
- Repeat till convergence
  - Expectation step (E-step) : estimate the conditional expectation k' = $E_{k|x,\theta'}$[log p(x, k|θ)] using current θ'.
  - Maximization step (M-step) : Estimate new ϴ given by maximizing the expectation of the log likelihood argmax$_\Theta$ $E_{k|x,\theta'}$[log p(x, k|θ)] ≅ argmax$_\Theta$ log p(x, k'|θ)

EM on GMM

- E-step
  - Set soft labels: $w_{n,j}$ = probability that nth sample comes from jth mixture p
  - Using Bayes rule
    - $p(k|x\,;\,\mu,\sigma,\phi) = \dfrac{p(x|k\,;\,\mu,\sigma,\phi)\,p(k;\,\mu,\sigma,\phi)}{p(x;\,\mu,\sigma,\phi)}$
    - $p(k|x\,;\,\mu,\sigma,\phi)$ is proportional to $p(x|k\,;\,\mu,\sigma,\phi)\,p(k;\,\phi)$

$$p(k_n = j|x_n;\phi,\mu,\Sigma) = \frac{\overbrace{p(x_n;\mu_j,\sigma_j)}^{N(\mu_j,\sigma_j)}\overbrace{p(k_n = j;\phi)}^{\phi_j}}{\Sigma_l p(x_n;\mu_l,\sigma_l)p(k_n = l;\phi)}$$

EM on GMM

- M-step (hard labels)

$$\phi_j = \frac{1}{N}\Sigma_{n=1}^{N} 1(k_n = j)$$

$$\mu_j = \frac{\Sigma_{n=1}^{N} 1(k_n = j) x_n}{\Sigma_{n=1}^{N} 1(k_n = j)}$$

$$\sigma_j^2 = \frac{\Sigma_{n=1}^{N} 1(k_n = j)(x_n - \mu_j)^2}{\Sigma_{n=1}^{N} 1(k_n = j)}$$

EM on GMM

- M-step (soft labels)

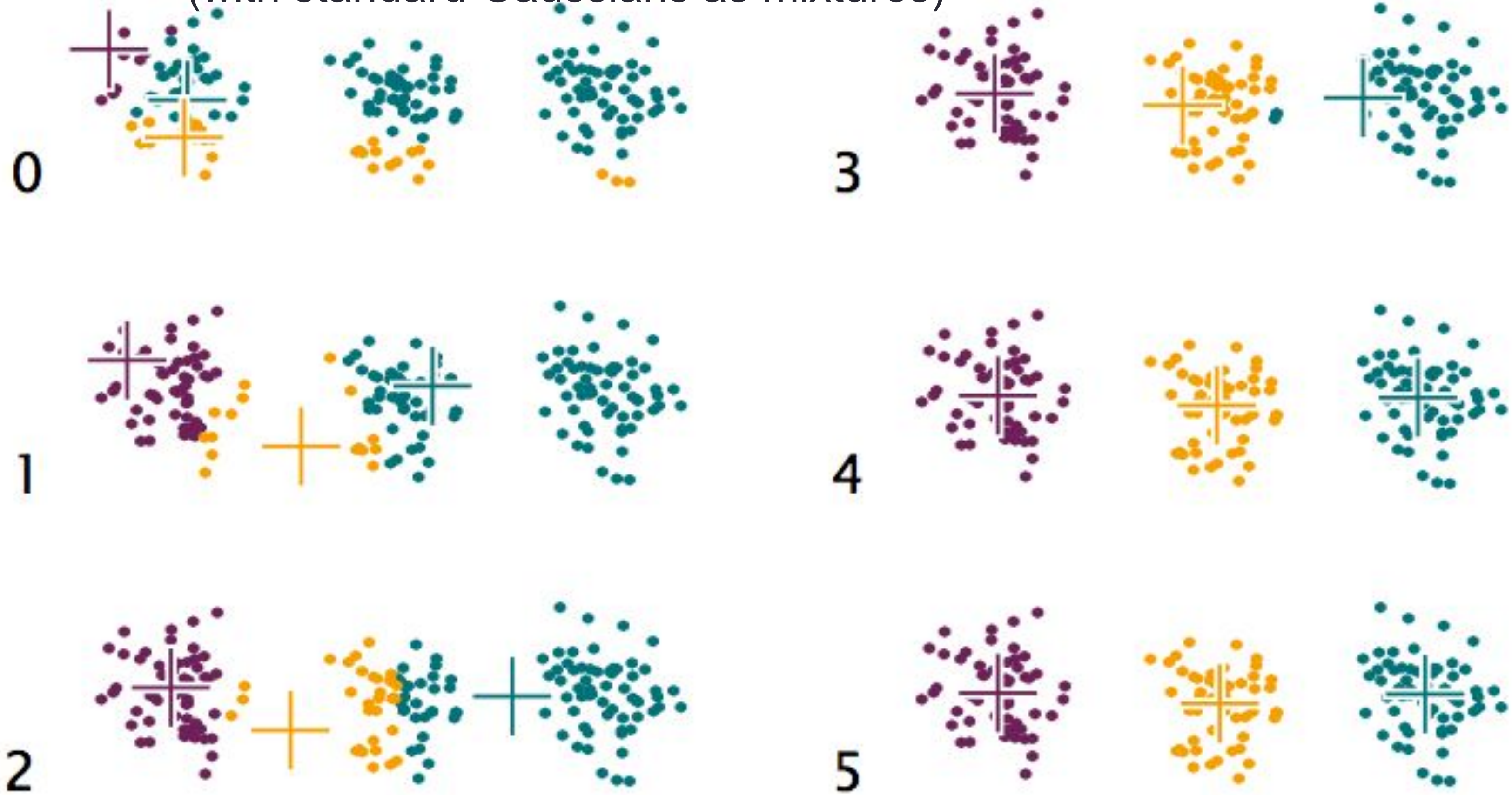
$$\phi_j = \frac{1}{N} \Sigma_{n=1}^{N} w_{n,j}$$

$$\mu_j = \frac{\Sigma_{n=1}^{N} w_{n,j} x_n}{\Sigma_{n=1}^{N} w_{n,j}}$$

$$\sigma_j^2 = \frac{\Sigma_{n=1}^{N} w_{n,j} (x_n - \mu_j)^2}{\Sigma_{n=1}^{N} w_{n,j}}$$

K-mean vs EM

EM on GMM can be considered as EM with soft labels
(with standard Gaussians as mixtures)



K-mean clustering

- Task: cluster data into groups
- K-mean algorithm
  - Initialization: Pick K data points as cluster centers
  - Assign: Assign data points to the closest centers
  - Update: Re-compute cluster center
  - Repeat: Assign and Update

EM algorithm for GMM

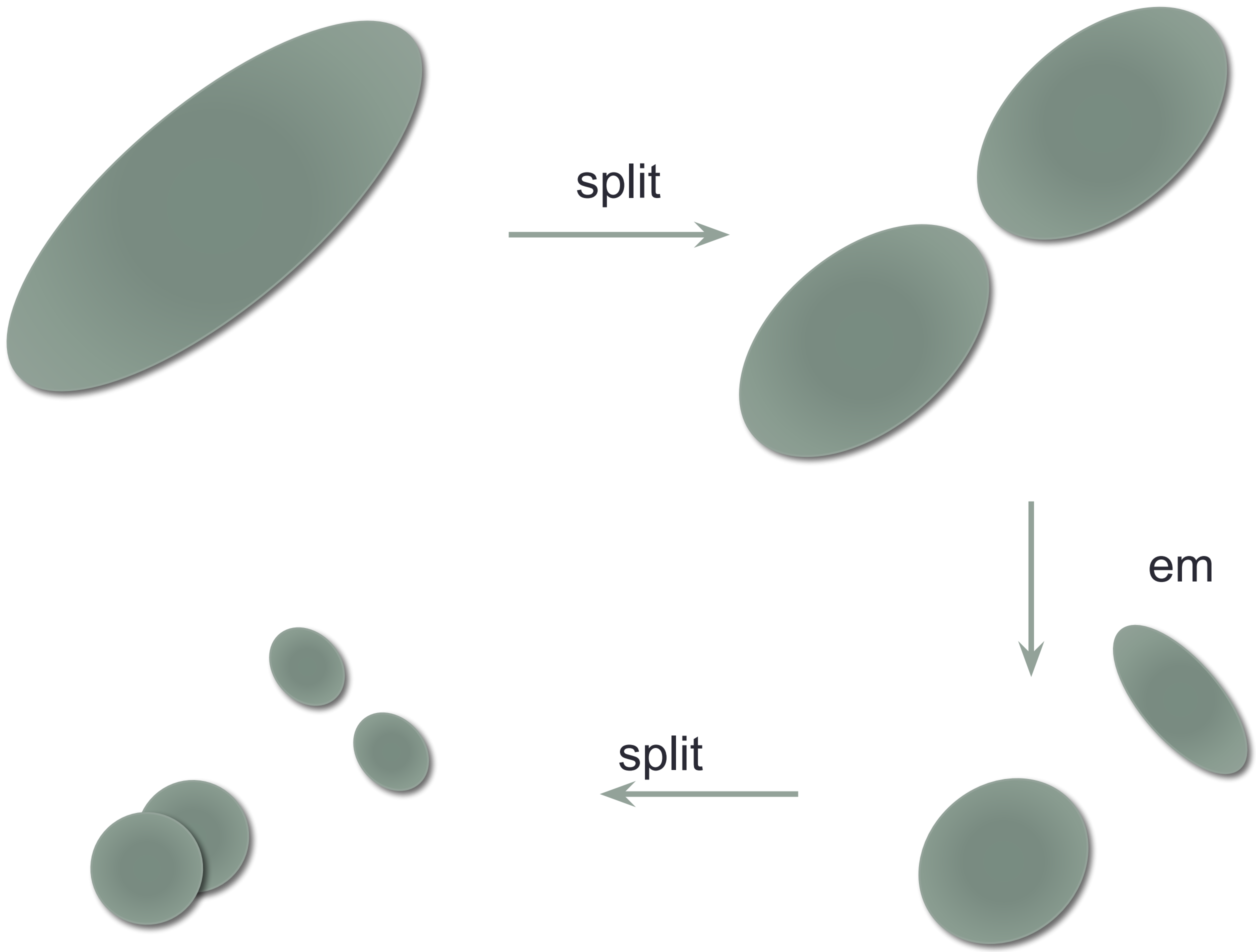
- Task: cluster data into Gaussians
- EM algorithm
  - Initialization: Randomly initialize parameters Gaussians
  - Expectation: Assign data points to the closest Gaussians
  - Maximization: Re-compute Gaussians parameters according to assigned data points
  - Repeat: Expectation and Maximization

- Note: assigning data points is actually a soft assignment (with probability)

EM/GMM notes

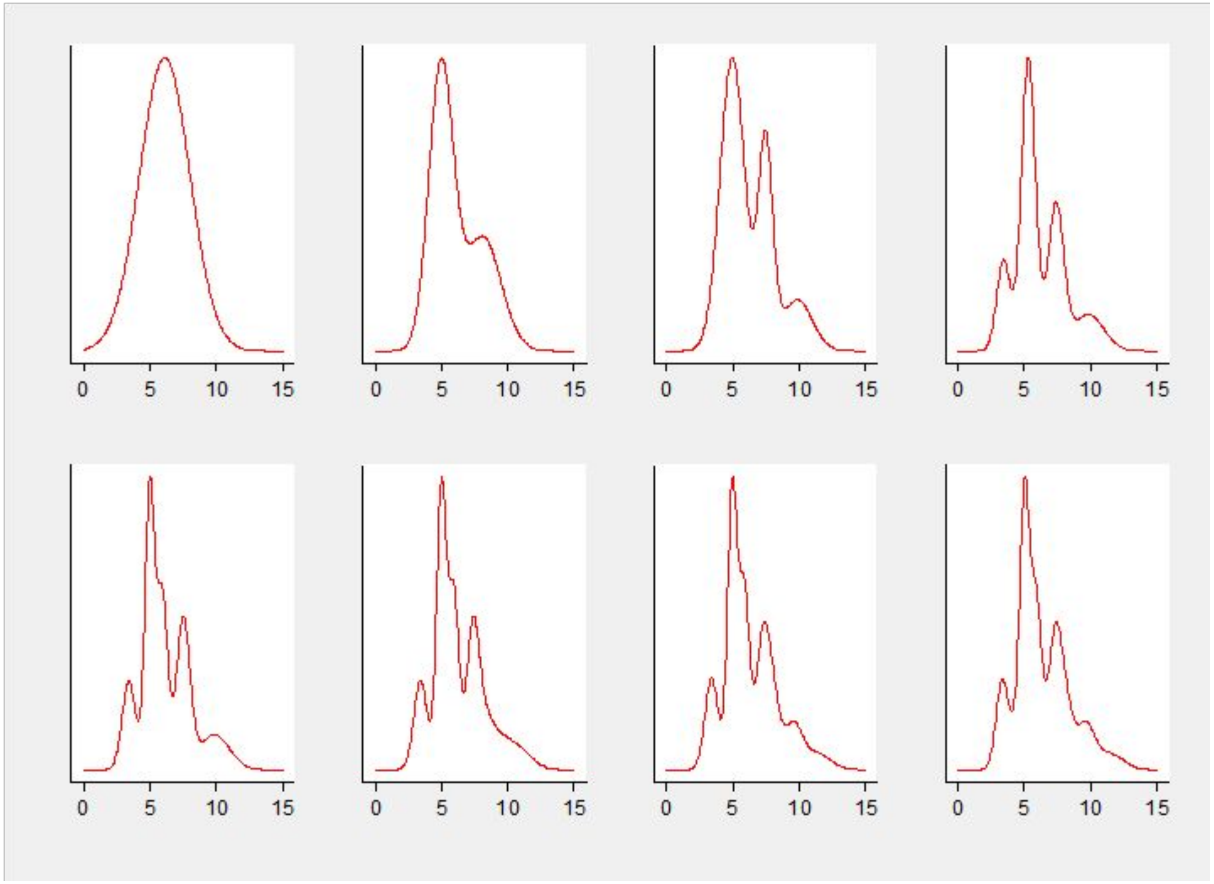
- Converges to local maxima (maximizing likelihood)
  - Just like k-means, need to try different initialization points
- EM always improve the likelihood for each iteration
  - Stops EM when likelihood changes < threshold
- Just like k-means some centroid can get stuck with one sample point and no longer moves
  - For EM on GMM this cause variance to go to 0…
    - Introduce variance floor (minimum variance a Gaussian can have)
- Tricks to avoid bad local maxima
  - Starts with 1 Gaussian
  - Split the Gaussians according to the direction of maximum variance
  - Repeat until arrive at k Gaussians
  - Does not guarantee global maxima but works well in practice

Gaussian splitting



Picking the amount of Gaussians

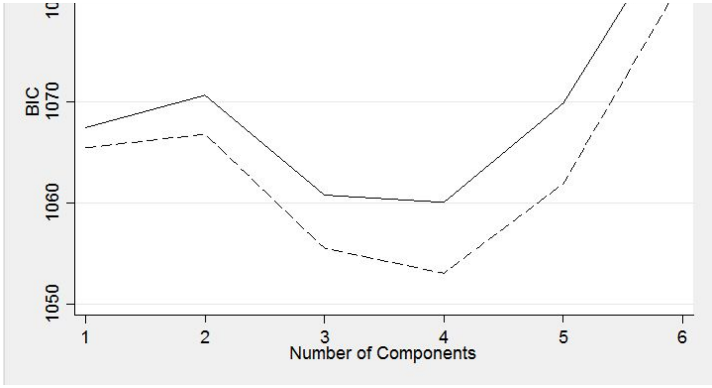
- As we increase K, the likelihood will keep increasing
- More mixtures -> more parameters -> overfits



http://staffblogs.le.ac.uk/bayeswithstata/2014/05/22/mixture-models-how-many-components/

Picking the amount of Gaussians

- Need a measure of goodness (like Elbow method in k-mean)
- Bayesian Information Criterion (BIC)
- Penalize the log likelihood from the data by the amount of parameters in the model
  - -2 log L + t log (n)
  - t = number of parameters in the model
  - n = number of data points
- We want to mimimize BIC



BIC is bad use cross validation!

- BIC is bad use cross validation!
- BIC is bad use cross validation!
- BIC is bad use cross validation!
- Test on the goal of your model

Latent variables?

EM is all about problem formulation. <u>You can solve the same task with different formulations.</u>

Latent variable considerations

- Imaginary quantity meant to provide a simplified view of the process
  - GMM mixtures. Speech recognizer states. Customer segmentation.
- Real-world thing, but impossible to directly measure
  - Cause of a disease. Temperature of a star.
- Real-world thing, that is not measured because of noise/faulty sensors

Latent variables?

- Discrete latent variables: clusters/partitions data into subgroups
- Continuous latent variables: can be used for dimensionality reduction (factor analysis, etc)

EM on a simple example

- Grades in class $P(A) = 0.5$ $P(B) = 0.5\text{-}\theta$ $P(C) = \theta$
- We want to estimate θ from three known numbers
  - N_a N_b N_c
- Find the maximum likelihood estimate of θ

EM on a simple example

- Grades in class $P(A) = 0.5$ $P(B) = 0.5-\theta$ $P(C) = \theta$
- We want to estimate θ from ONE known number
  - N_c (we also know N the total number of students)
- Find θ using EM

EM usage examples

Image segmentation with GMM EM

- D - {r,g,b} value at each pixel
- Latent : segment where each pixel comes from
- Hyperparameters: number of mixtures (K), initial values

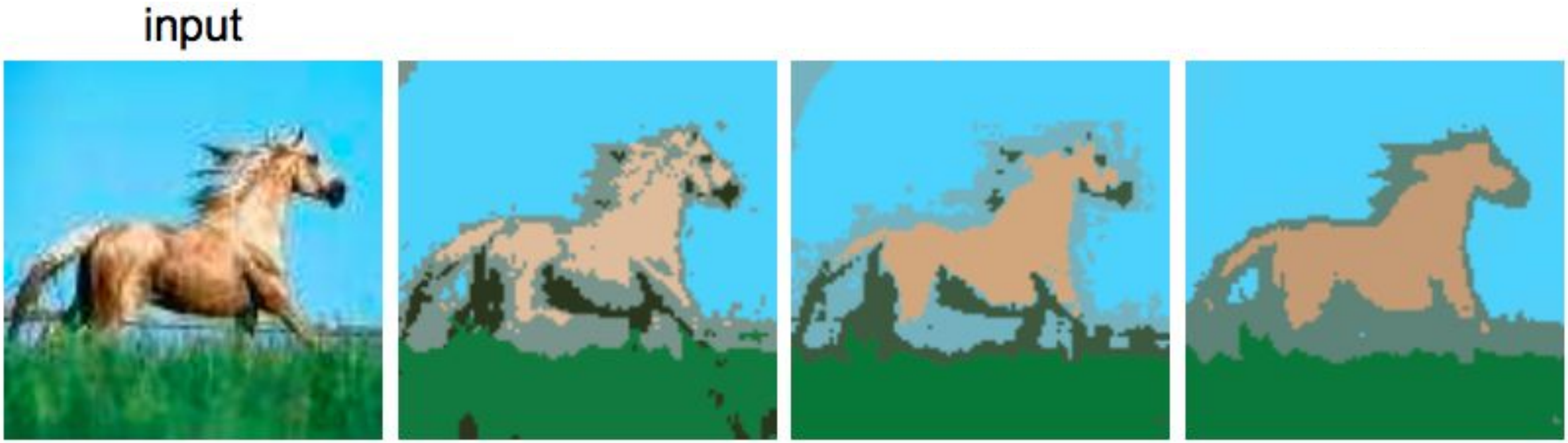


Image segmentation with GMM EM

(a) (b) (c)

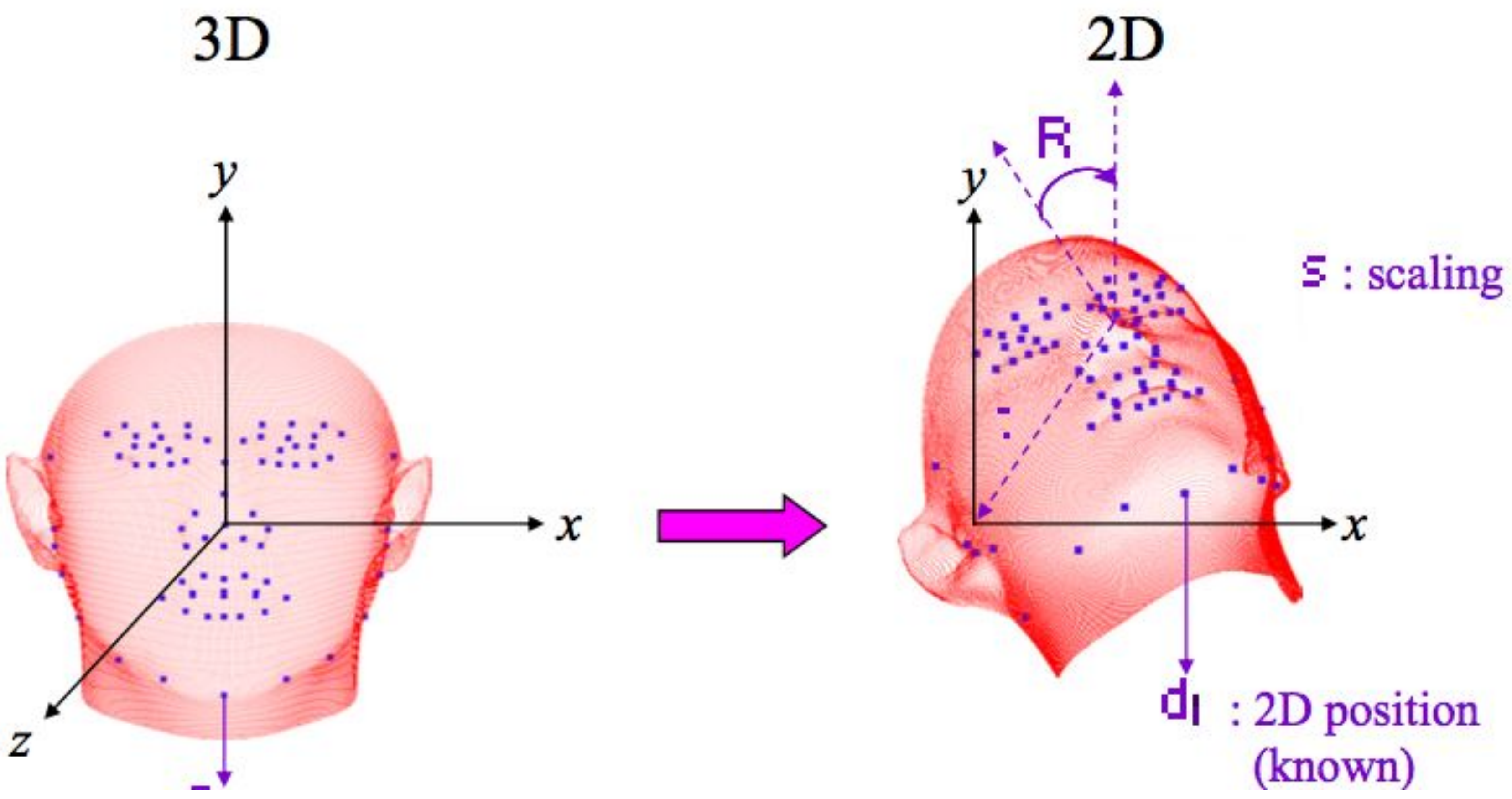
Fig. 1. Original images: (a) flower, (b) tiger, (c) bear

Fig. 2. Segmentation results ($M = 2$)

Fig. 3. Segmentation results ($M = 5$)

Zhaoxia Fu Color Image Segmentation Using Gaussian Mixture Model and EM Algorithm

Face pose estimation (estimate 3d coordinates from 2d picture)



https://www.researchgate.net/publication/2489713_Estimating_3D_Facial_Pose_using_the_EM_Algorithm

Language modeling


THE UNITED STATES CONSTITUTION

We the People of the United States, in Order to form a more perfect Union, establish Justice, insure domestic Tranquility, provide for the common defense, promote the general Welfare, and secure the Blessings of Liberty to ourselves and our Posterity, do ordain and establish this Constitution for the United States of America.

Article I.

Section 1.
All legislative Powers herein granted shall be vested in a Congress of the United States, which shall consist of a Senate and House of Representatives.

Section 2.
Clause 1: The House of Representatives shall be composed of Members chosen every second Year by the People of the several States, and the Electors in each State shall have the Qualifications requisite for Electors of the most numerous Branch of the State Legislature.

Clause 2: No Person shall be a Representative who shall not have attained to the Age of twenty-five Years, and been seven Years a Citizen of the United States, and who shall not, when elected, be an Inhabitant of that State in which he shall be chosen.

Clause 3: Representatives and direct Taxes shall be apportioned among the several States which may be included within this Union, according to their respective Numbers, which shall be determined by adding to the whole Number of free Persons, including those bound to Service for a Term of Years, and excluding Indians not taxed, three fifths of all other Persons. The actual

Latent variable: Topic
P(word|topic)

For examples: see Probabilistic latent semantic analysis

Summary

- GMM
  - Mixture of Gaussians
- EM
  - Expectation
  - Maximization

More info and exact proofs

https://www.cs.utah.edu/~piyush/teaching/EM_algorithm.pdf