Expectation-Maximization

Léon Bottou

NEC Labs America

COS 424 – 3/9/2010

Agenda

Goals	Classification, clustering, regression, other.
Representation	Parametric vs. kernels vs. nonparametric Probabilistic vs. nonprobabilistic Linear vs. nonlinear Deep vs. shallow
Capacity Control	Explicit: architecture, feature selection Explicit: regularization, priors Implicit: approximate optimization Implicit: bayesian averaging, ensembles
Operational Considerations	Loss functions Budget constraints Online vs. offline
Computational Considerations	Exact algorithms for small datasets. Stochastic algorithms for big datasets. Parallel algorithms.

Summary

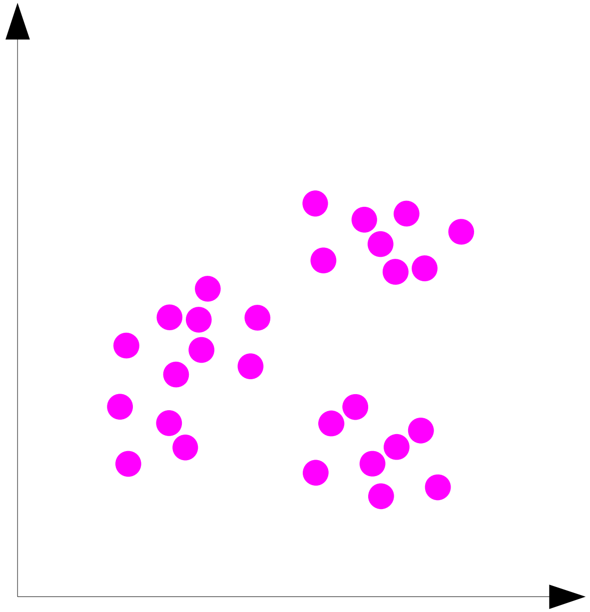
Expectation Maximization

– Convenient algorithm for certain Maximum Likelihood problems.
– Viable alternative to Newton or Conjugate Gradient algorithms.
– More fashionable than Newton or Conjugate Gradients.
– Lots of extensions: variational methods, stochastic EM.

Outline of the lecture

1. Gaussian mixtures.
2. More mixtures.
3. Data with missing values.

Simple Gaussian mixture



Clustering via density estimation.

– Pick a parametric model $\mathbb{P}_\theta(X)$.

– Maximize likelihood.

Parametric model

– There are K components

– To generate an observation:

a.) pick a component k

with probabilities $\lambda_1 \dots \lambda_K$ with $\sum_k \lambda_k = 1$.

b.) generate x from component k

with probability $\mathcal{N}(\mu_i, \sigma)$.

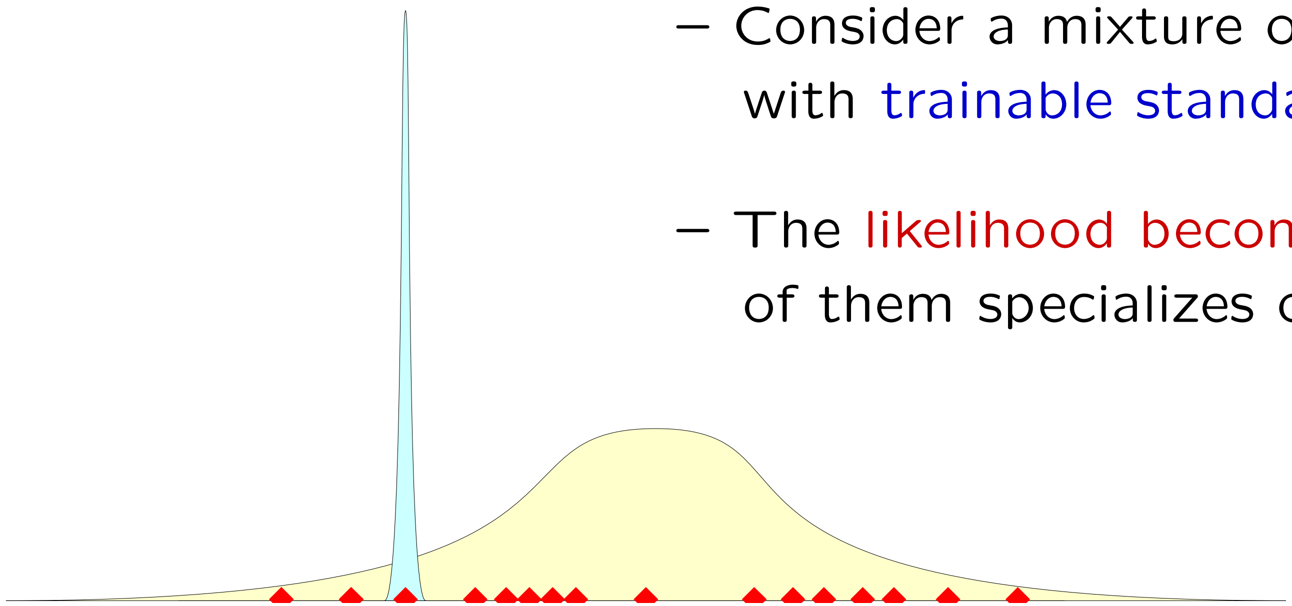
Simple GMM: Standard deviation σ known and constant.

– What happens when σ is a trainable parameter?

– Different σ_i for each mixture component?

– Covariance matrices Σ instead of scalar standard deviations ?

When Maximum Likelihood fails



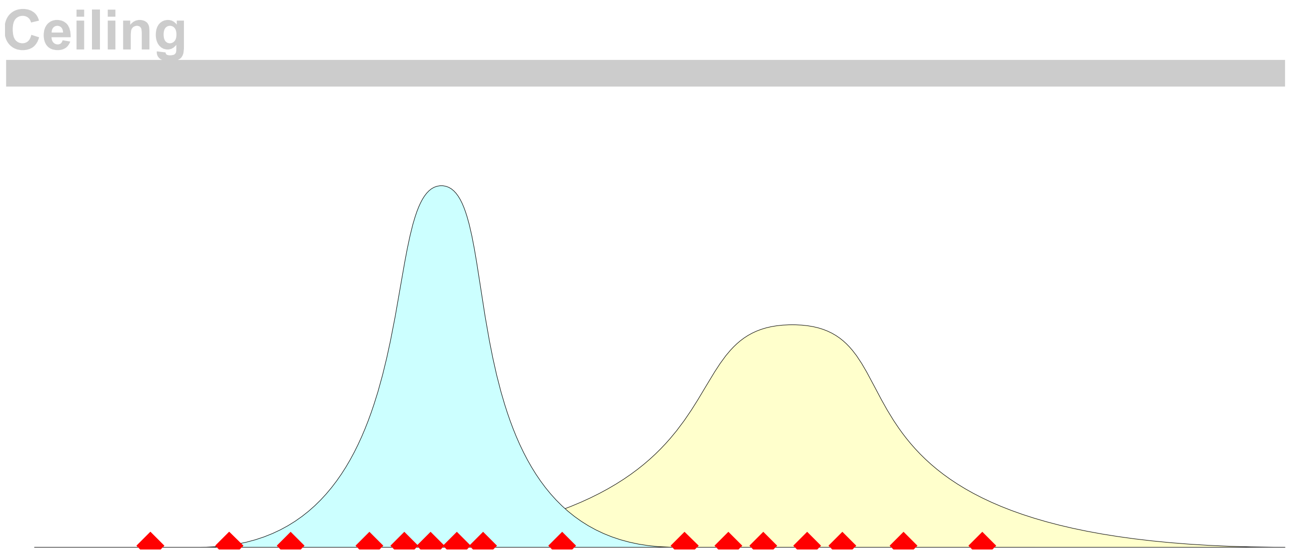
– Consider a mixture of two Gaussians with trainable standard deviations.

– The likelihood becomes infinite when one of them specializes on a single observation.

– MLE works for all discrete probabilistic models and for some continuous probabilistic models.
– This simple Gaussian mixture model is not one of them.
– People just ignore the problem and get away with it.

Why ignoring the problem does work ?

Explanation 1 – The GMM likelihood has many local maxima.



– Unlike discrete distributions, densities are not bounded.
A ceiling on the densities theoretically fixes the problem.
Equivalently: enforcing a minimal standard deviation that prevents Gaussians to specialize on a single observation...

– The singularity lies in a narrow corner of the parameter space.
Optimization algorithms cannot find it!.

Why ignoring the problem does work ?

Explanation 2 – There are no rules in the Wild West.

– We should not condition probabilities with respect to events with probability zero.

– With continuous probabilistic models, observations always have probability zero!

Expectation Maximization for GMM

– We only observe the $x_1, x_2, \dots$.
– Some models would be very easy to optimize if we knew which mixture components $y_1, y_2, \dots$ generates them.

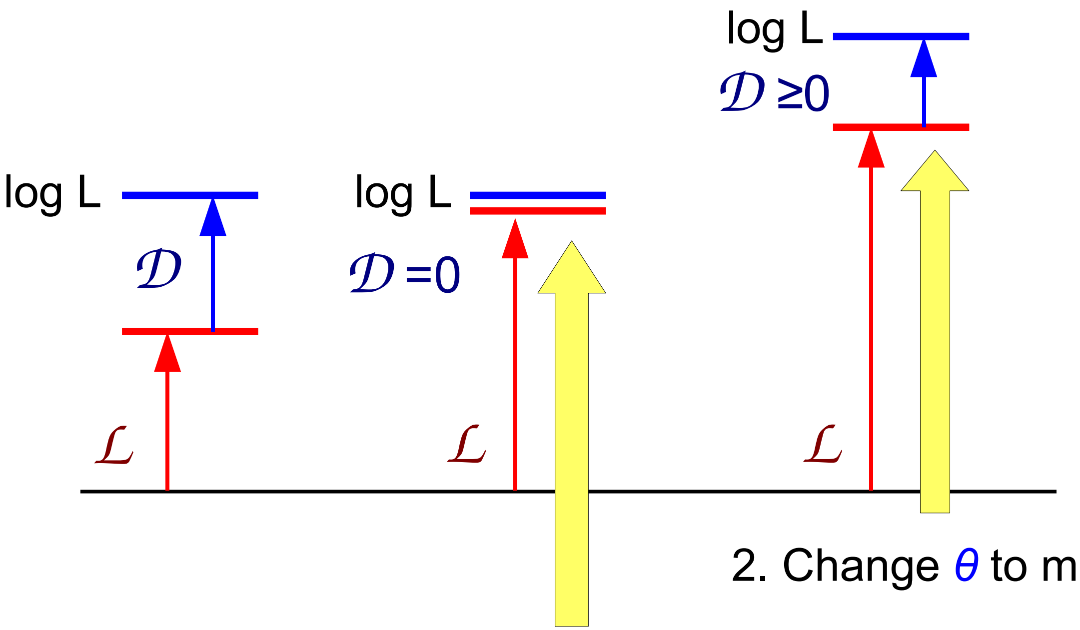
Decomposition

– For a given X, guess a distribution $Q(Y|X)$.
– Regardless of our guess, $\log L(\theta) = \mathcal{L}(Q,\theta) + \mathcal{D}(Q,\theta)$

$$\mathcal{L}(Q,\theta) = \sum_{i=1}^{n}\sum_{y=1}^{K} Q(y|x_i)\log\frac{P_\theta(x_i|y)P_\theta(y)}{Q(y|x_i)} \qquad \text{Easy to maximize}$$

$$\mathcal{D}(Q,\theta) = \sum_{i=1}^{n}\sum_{y=1}^{K} Q(y|x_i)\log\frac{Q(y|x_i)}{P_\theta(y|x_i)} \qquad \text{KL divergence } D(Q_{Y|X}\|P_{Y|X})$$

Expectation-Maximization



2. Change θ to maximize $\mathcal{L}$. Meanwhile $\mathcal{D}$ can only increase.

1. Change Q to minimize $\mathcal{D}$ leaving log L unchanged.

E-Step: $q_{ik} \leftarrow \dfrac{\lambda_k}{\sqrt{|\Sigma_k|}}\, e^{-\frac{1}{2}\,(x_i-\mu_k)^\top \Sigma_k^{-1}(x_i-\mu_k)}$ *remark: normalization!.*

M-Step: $\mu_k \leftarrow \dfrac{\sum_i q_{ik}\, x_i}{\sum_i q_{ik}}$ $\quad \Sigma_k \leftarrow \dfrac{\sum_i q_{ik}(x_i-\mu_k)(x_i-\mu_k)^\top}{\sum_i q_{ik}}$ $\quad \lambda_k \leftarrow \dfrac{\sum_i q_{ik}}{\sum_{iy} q_{iy}}$

Implementation remarks

Numerical issues

– The q_{ik} are often very small and underflow the machine precision.

– Instead compute $\log q_{ik}$ and work with $\hat{q}_{ik} = q_{ik}\, e^{-\max_k(\log q_{ik})}$.

Local maxima

– The likelihood is highly non convex.

– EM can get stuck in a mediocre local maximum.

– This happens in practice. Initialization matters.

– On the other hand, the global maximum is not attractive either.

Computing the log likelihood

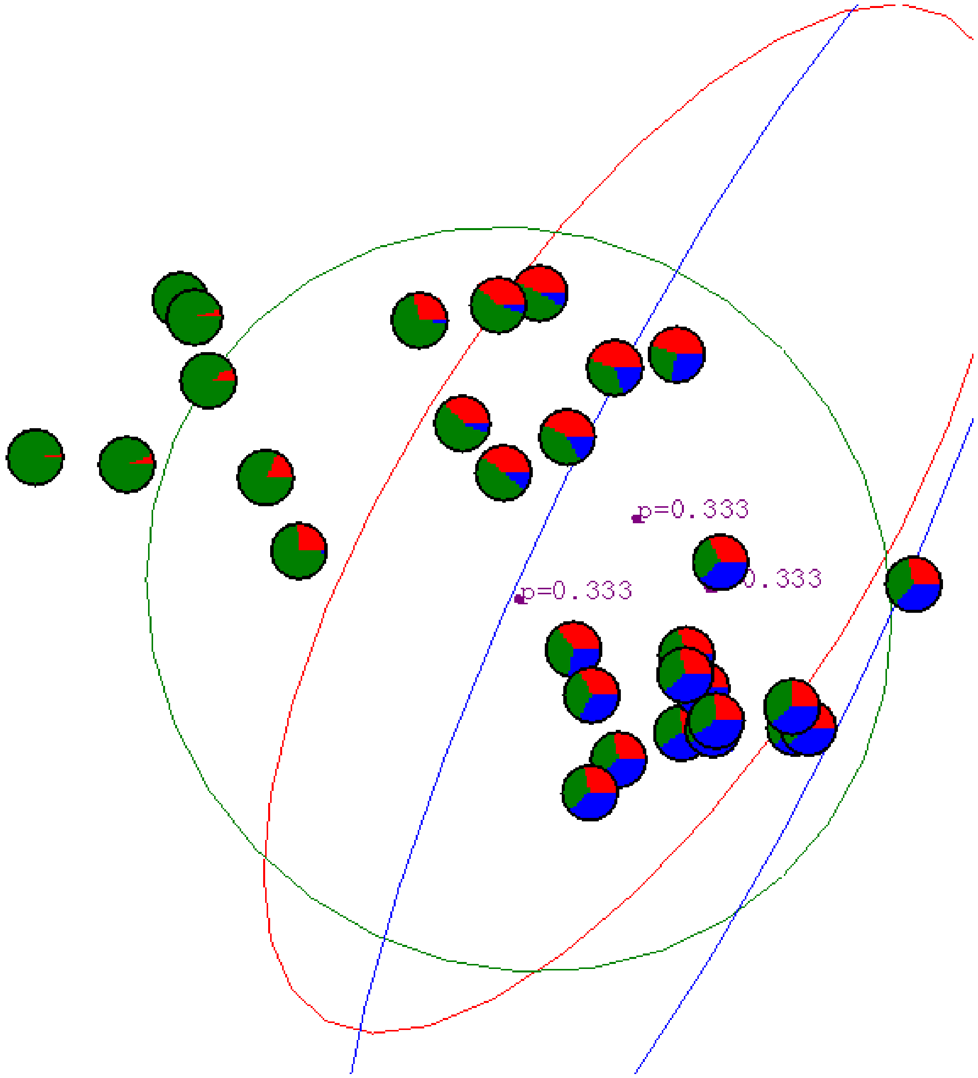
– Computing the log likelihood is useful to monitor the progress of EM.

– The best moment is after the E-step and before the M-step.

– Since $\mathcal{D} = 0$ it is sufficient to compute $\mathcal{L} - \mathcal{M}$.

EM for GMM

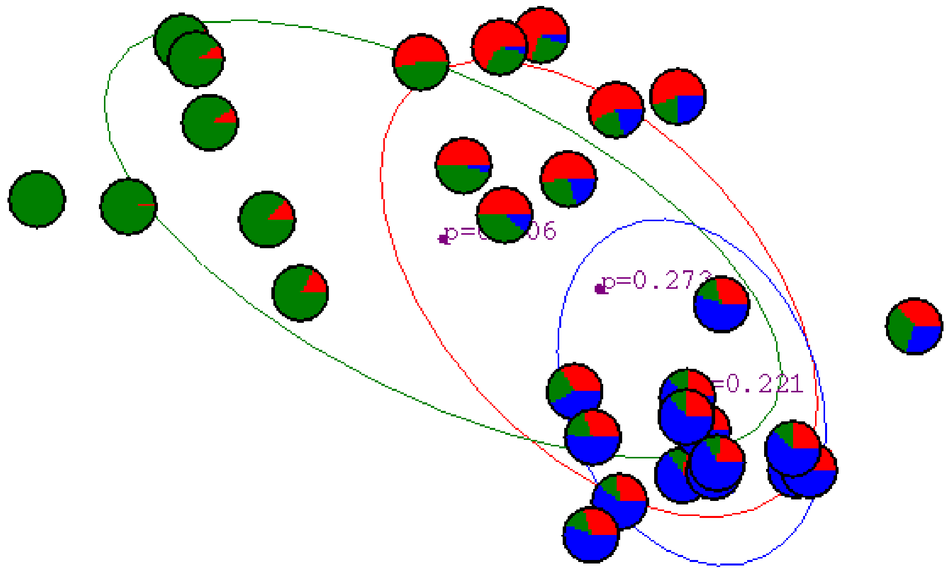
Start.



(Illustration from Andrew Moore's tutorial on GMM.)

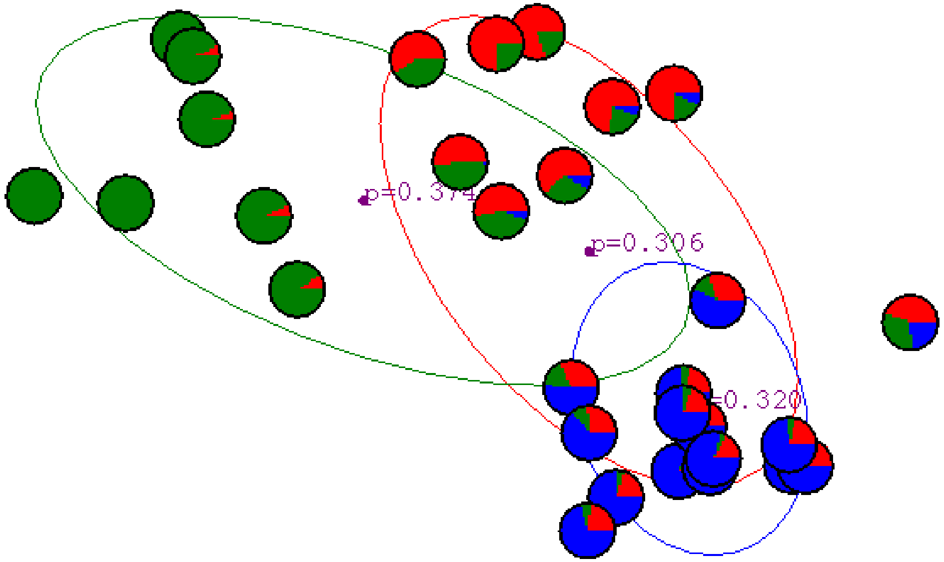
EM for GMM

After iteration #1.



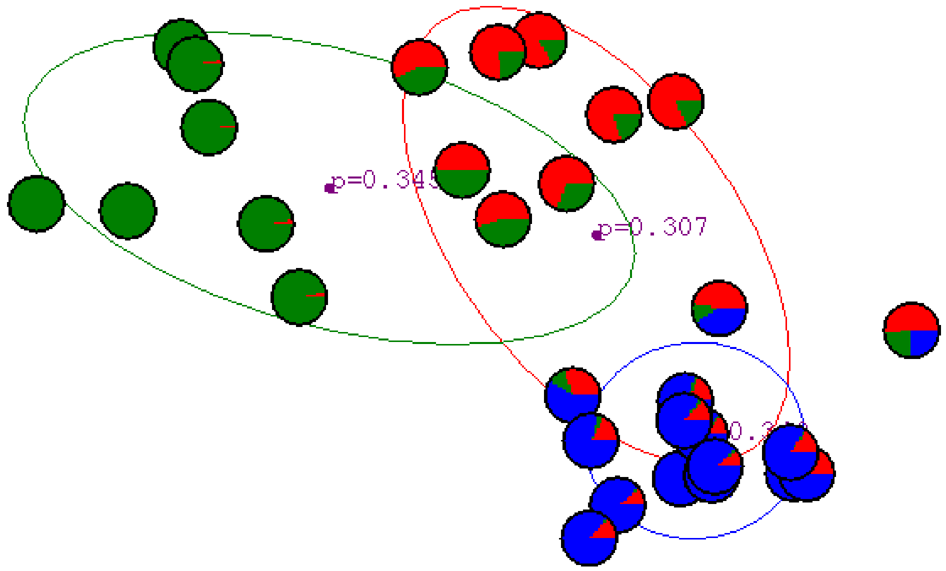
EM for GMM

After iteration #2.



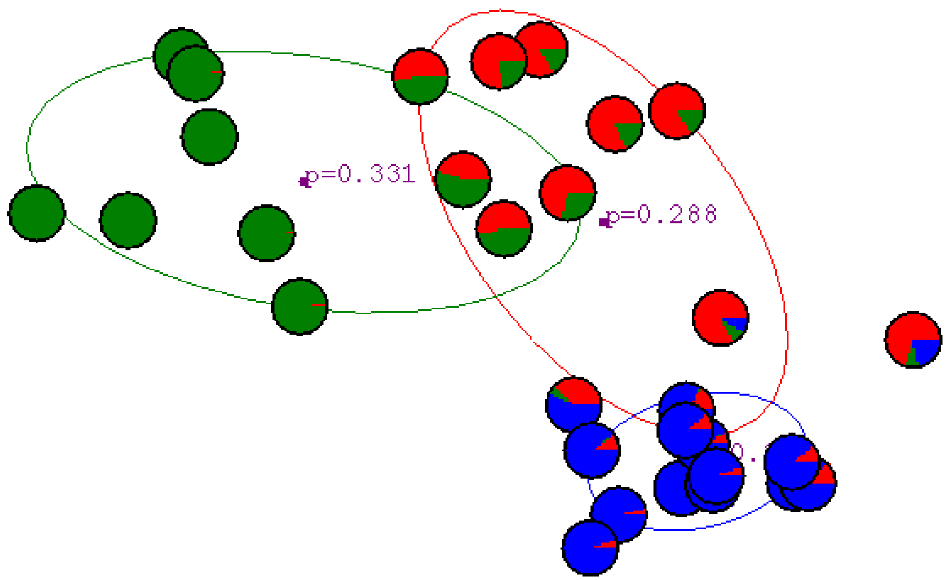
EM for GMM

After iteration #3.



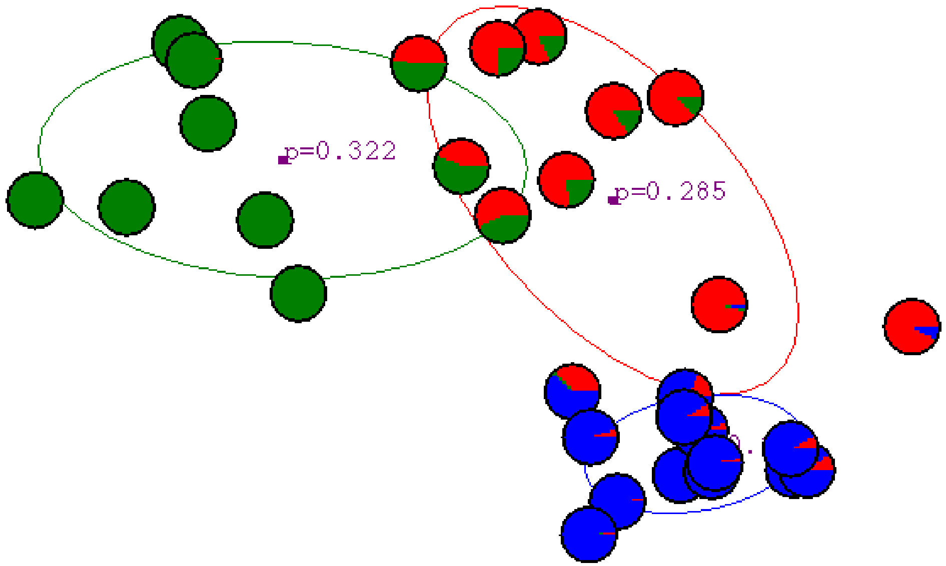
EM for GMM

After iteration #4.



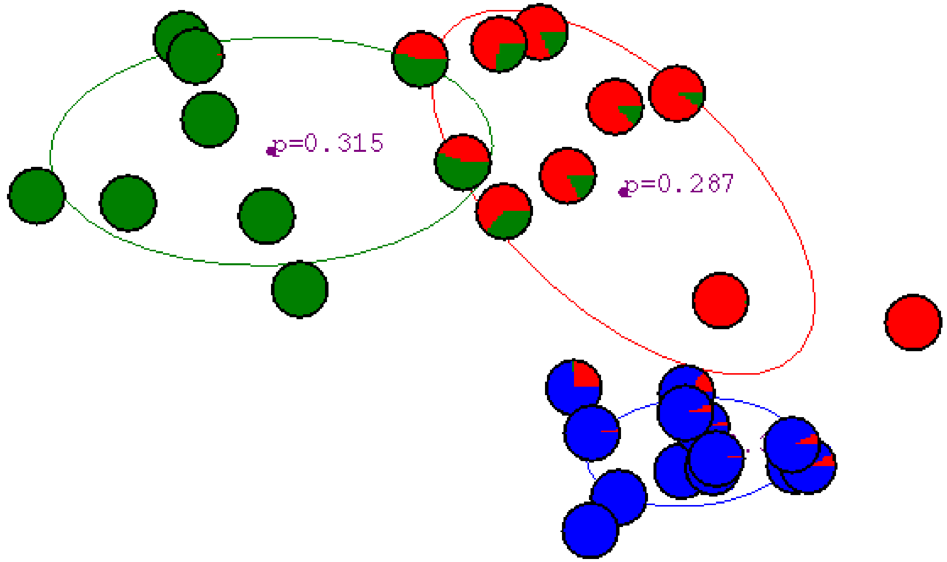
EM for GMM

After iteration #5.



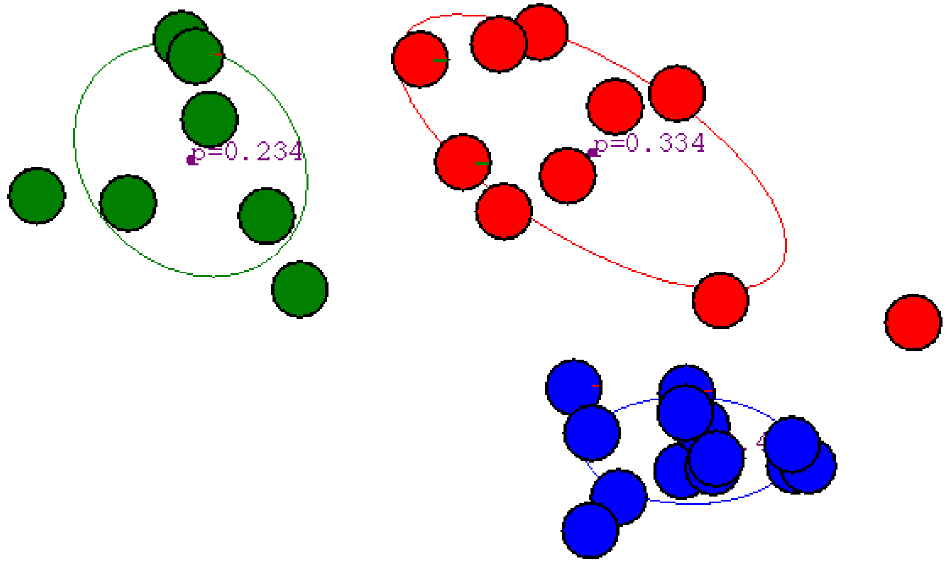
EM for GMM

After iteration #6.



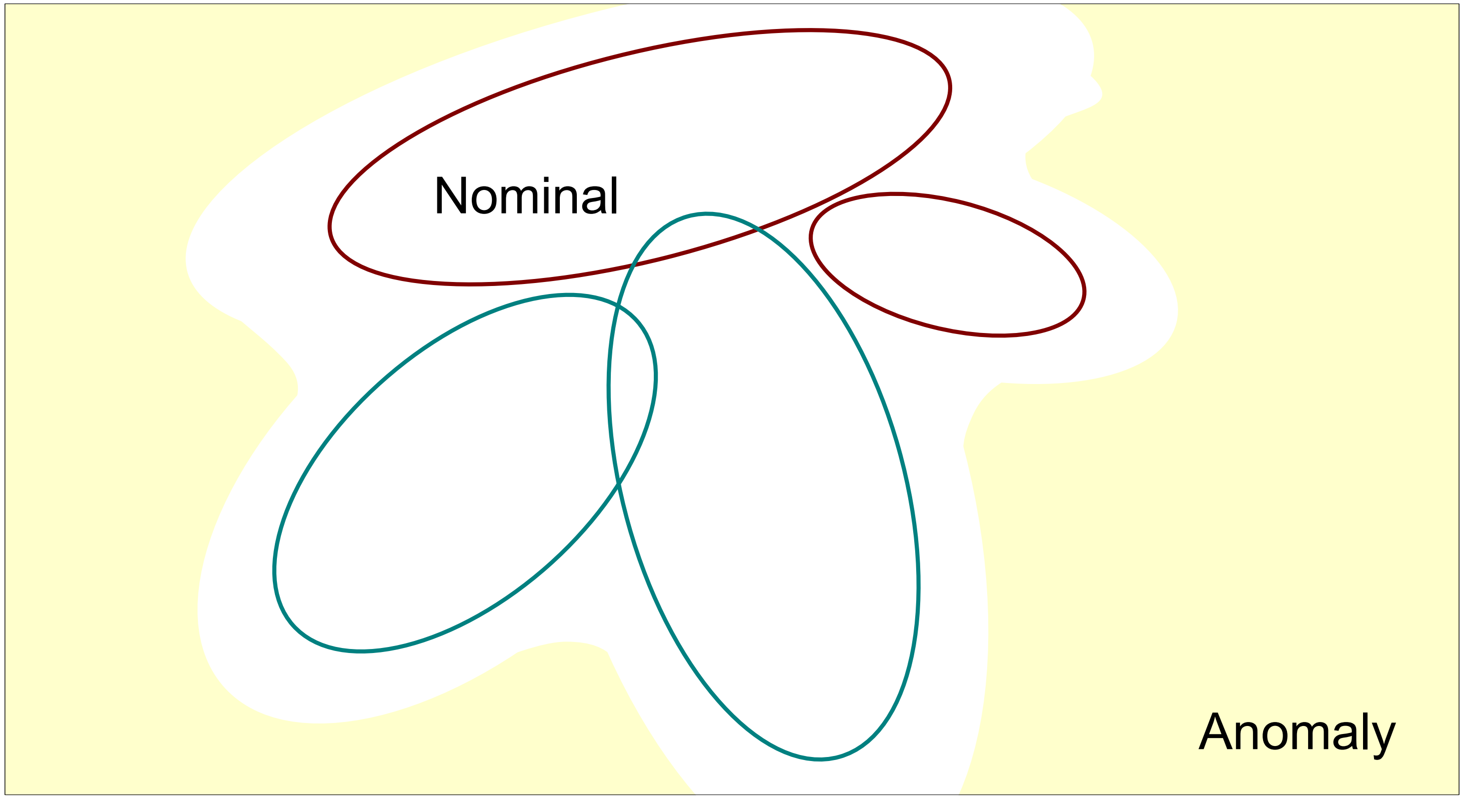
EM for GMM

After iteration #20



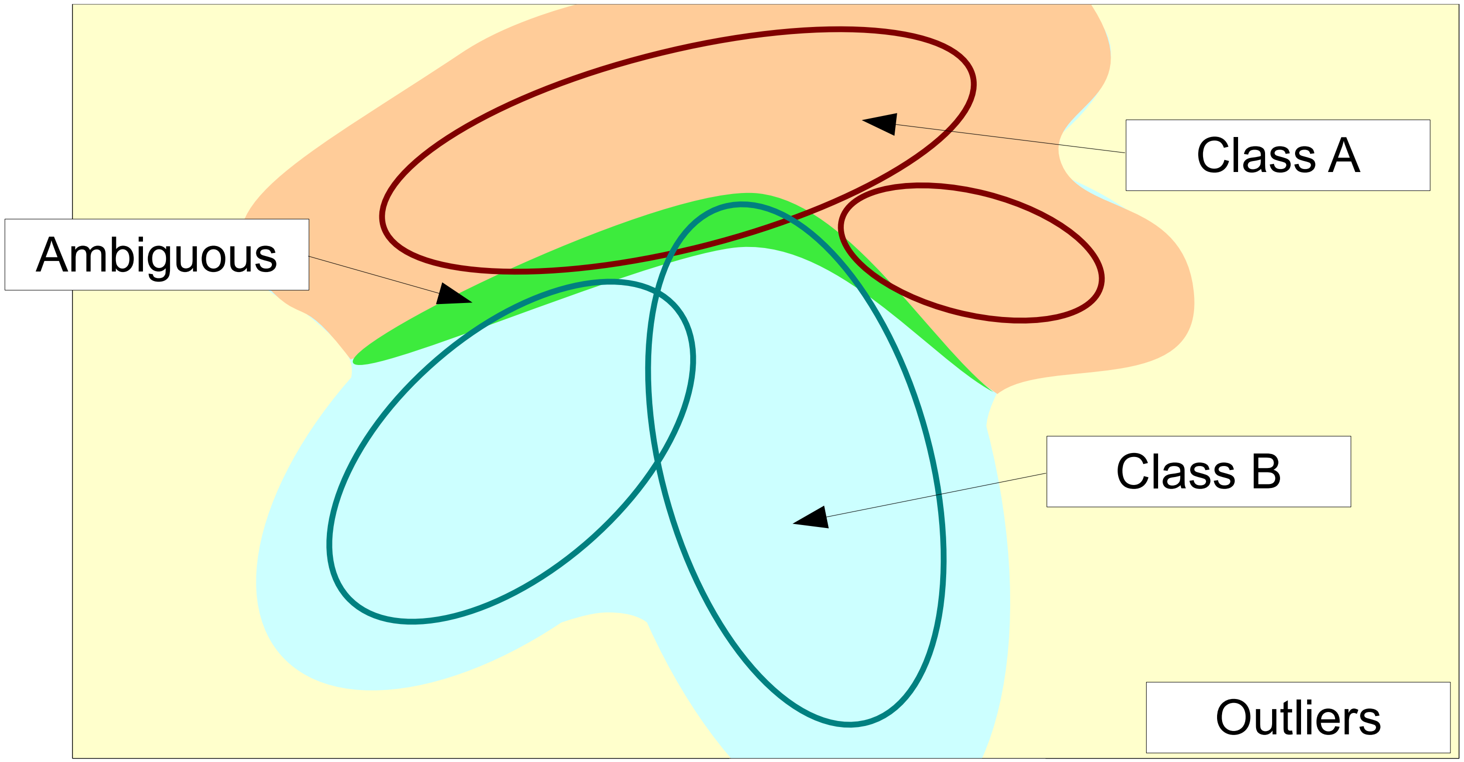
GMM for anomaly detection

1. Model $\mathbb{P}\{X\}$ with a GMM.
2. Declare anomaly when density fails below a threshold.



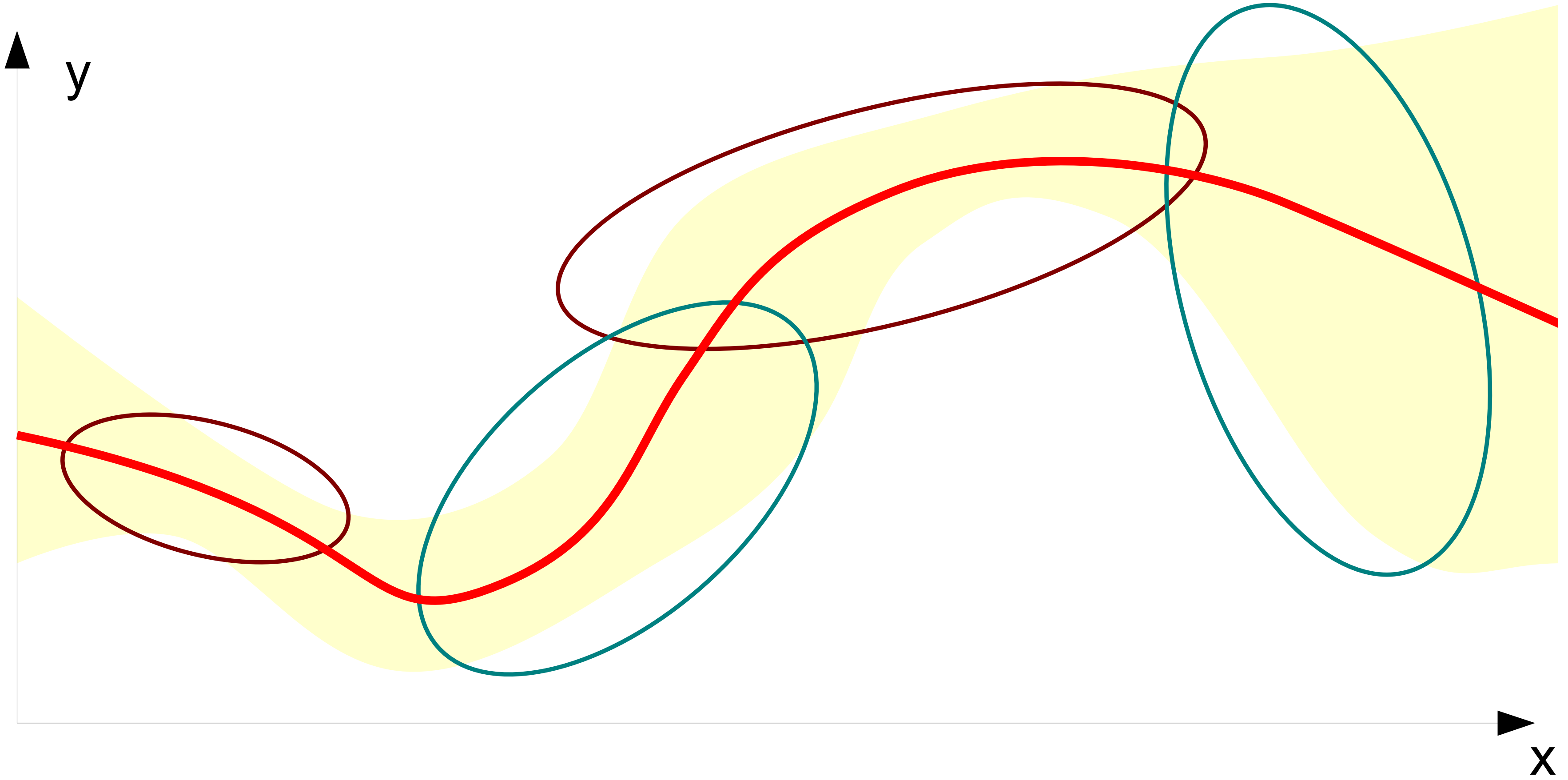
GMM for classification

1. Model $\mathbb{P}\{X \mid Y = y\}$ for every class with a GMM.
2. Calulate Bayes optimal decision boundary.
3. Possibility to detect outliers and ambiguous patterns.



GMM for regression

1. Model $\mathbb{P}\{X, Y\}$ with a GMM.
2. Compute $f(x) = \mathbb{E}[Y \mid X = x]$.



The price of probabilistic models

Estimating densities is nearly impossible!

– A GMM with many components is very flexible model.
– Nearly as demanding as a general model.

Can you trust the GMM distributions?

– Maybe in very low dimension. . .
– Maybe when the data is abundant. . .

Can you trust decisions based on the GMM distribution?

– They are often more reliable than the GMM distributions themselves.
– Use cross-validation to check!

Alternatives?

– Directly learn the decision function!
– Use cross-validation to check!.

More mixture models

We can make mixtures of anything.

Bernoulli mixture

Example: Represent a text document by D binary variables indicating the presence or absence of word $t = 1 \dots D$.

– Base model: model each word independently with a Bernoulli.

– Mixture model: see next slide.

Non homogeneous mixtures

It is sometimes useful to mix different kinds of distributions.

Example: model how long a patient survives after a treatment.

– One component with thin tails for the common case.

– One component with thick tails for patients cured by the treatment.

Bernoulli mixture

Consider D binary variables $\mathbf{x} = (x_1, \ldots, x_D)$.
Each x_i independently follows a Bernoulli distribution $B(\mu_i)$.

$$P_{\boldsymbol{\mu}}(\mathbf{x}) = \prod_{i=1}^{D} \mu_i^{x_i} (1-\mu_i)^{1-x_i}$$

Mean	$\boldsymbol{\mu}$
Covariance	$\mathrm{diag}[\mu_i(1-\mu_i)]$

Now let's consider a mixture of such distributions.
The parameters are $\theta = (\lambda_1, \boldsymbol{\mu}_1, \ldots \lambda_k, \boldsymbol{\mu}_k)$ with $\sum_k \lambda_k = 1$.

$$P_\theta(\mathbf{x}) = \sum_{k=1}^{K} \lambda_k P_{\boldsymbol{\mu}_k}(\mathbf{x}_i)$$

Mean	$\sum_k \lambda_k \boldsymbol{\mu}_k$
Covariance	*no longer diagonal!*

Since the covariance matrix is no longer diagonal,
the mixture models dependencies between the x_i.

EM for Bernoulli mixture

We are given a dataset $\mathbf{x} = \mathbf{x}_1, \dots, \mathbf{x}_n$.

The log likelihood is $\log L(\theta) = \displaystyle\sum_{i=1}^{n} \log \sum_{i=1}^{k} \lambda_k P_{\boldsymbol{\mu}_k}(\mathbf{x}_i)$

Let's derive an EM algorithm.

Variable $Y = y_1, \dots, y_n$ says which component generates X.

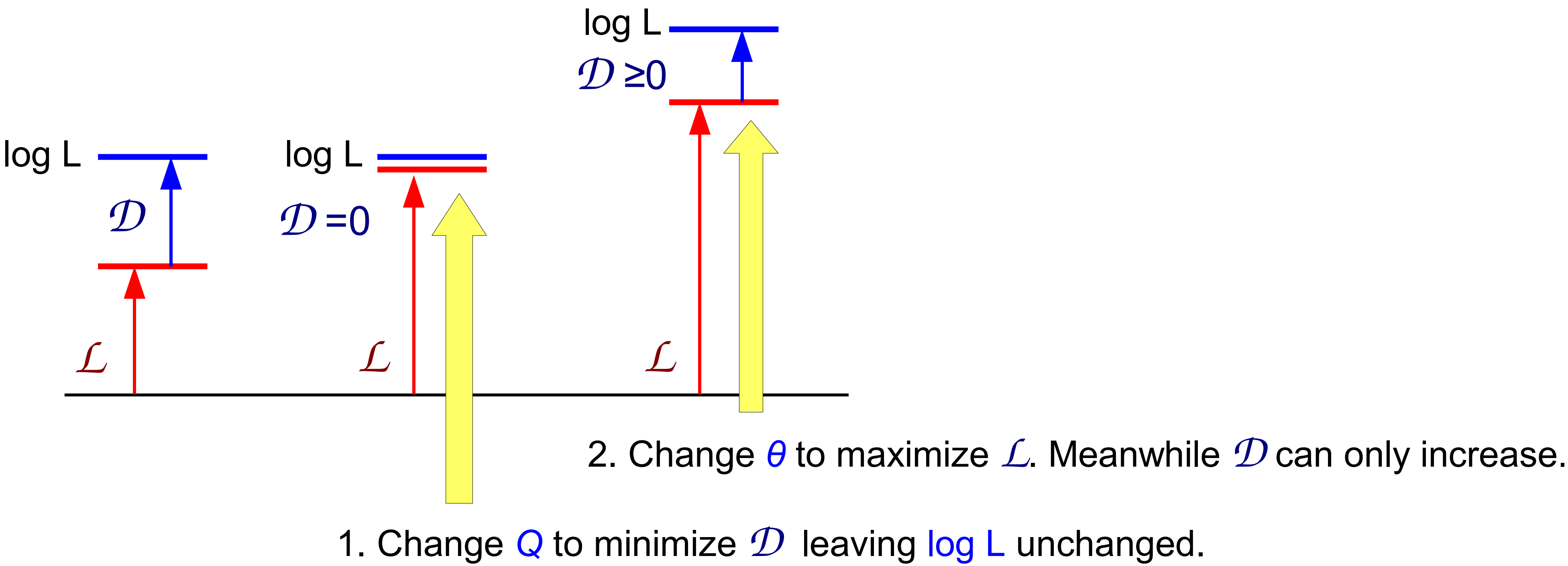
Maximizing the likelihood would be easy if we were observing the Y.

So let's just guess Y with distribution $Q(Y = y | X = x_i) \propto q_{iy}$.

Decomposition: $\log L(\theta) = \mathcal{L}(Q, \theta) + \mathcal{D}(Q, \theta)$,

with the usual definitions (slide 8.)

EM for a Bernoulli mixture



E-Step: $q_{ik} \leftarrow \lambda_k \, P_{\boldsymbol{\mu}_k}(\mathbf{x}_i)$ *remark: normalization!.*

M-Step: $\boldsymbol{\mu}_k \leftarrow \dfrac{\sum_i q_{ik}\,\mathbf{x}_i}{\sum_i q_{ik}} \qquad \lambda_k \leftarrow \dfrac{\sum_i q_{ik}}{\sum_{iy} q_{iy}}$

Data with missing values

"Fitting my probabilistic model would be so easy without missing values."

mpg	cyl	disp	hp	weight	accel	year	name
15.0	8	350.0	165.0	3693	11.5	70	buick skylark 320
18.0	8	318.0	150.0	3436	11.0	70	plymouth satellite
15.0	8	429.0	198.0	4341	10.0	70	ford galaxie 500
14.0	8	454.0	n/a	4354	9.0	70	chevrolet impala
15.0	8	390.0	190.0	3850	8.5	70	amc ambassador dpl
n/a	8	340.0	n/a	n/a	8.0	70	plymouth cuda 340
18.0	4	121.0	112.0	2933	14.5	72	volvo 145e
22.0	4	121.0	76.00	2511	18.0	n/a	volkswagen 411
21.0	4	120.0	87.00	2979	19.5	72	peugeot 504
26.0	n/a	96.0	69.00	2189	18.0	72	renault 12
22.0	4	122.0	86.00	n/a	16.0	72	ford pinto
28.0	4	97.0	92.00	2288	17.0	72	datsun 510
n/a	8	440.0	215.0	4735	n/a	73	chrysler new yorker

EM for missing values

"Fitting my probabilistic model would be so easy without missing values."

This magic sentence suggests EM

– Let $X = x_1, x_2, \ldots, x_n$ be the observed values on each row.

– Let $Y = y_1, y_2, \ldots, y_n$ be the missing values on each row.

Decomposition

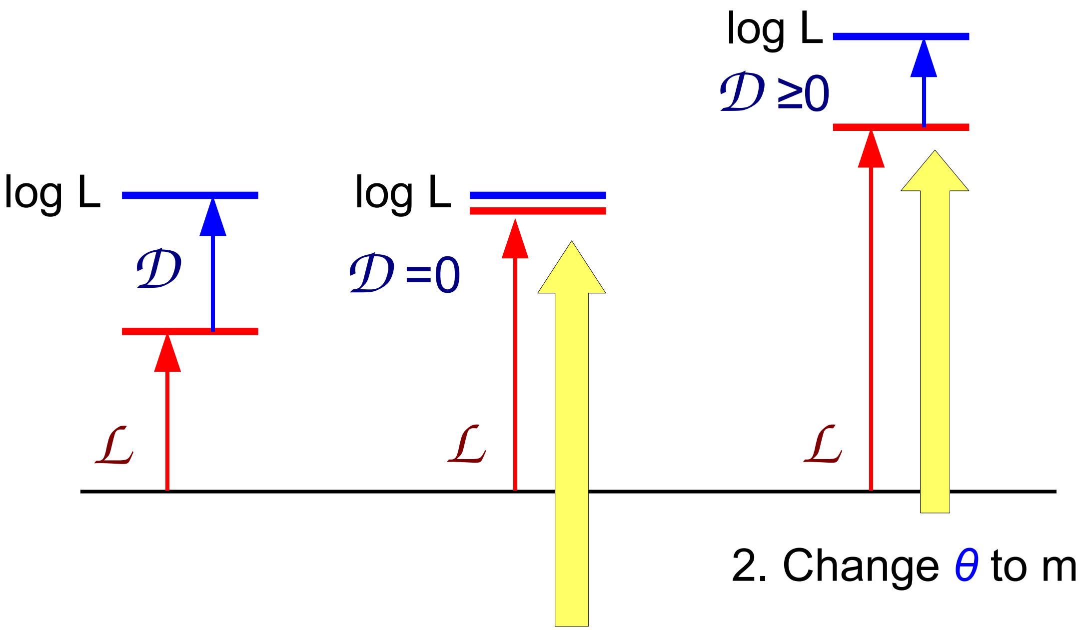
– Guess a distribution $Q_\lambda(Y|X)$.

– Regardless of our guess, $\log L(\theta) = \mathcal{L}(\lambda, \theta) + \mathcal{D}(\lambda, \theta)$

$$\mathcal{L}(\lambda, \theta) = \sum_{i=1}^{n} \sum_{y} Q_\lambda(y|x_i) \log \frac{P_\theta(x_i, y)}{Q_\lambda(y|x_i)} \quad \text{Easy to maximize}$$

$$\mathcal{D}(\lambda, \theta) = \sum_{i=1}^{n} \sum_{y} Q_\lambda(y|x_i) \log \frac{Q_\lambda(y|x_i)}{P_\theta(y|x_i)} \quad \text{KL divergence } D(Q_{Y|X} \| P_{Y|X})$$

EM for missing values



2. Change θ to maximize $\mathcal{L}$. Meanwhile $\mathcal{D}$ can only increase.

1. Change Q to minimize $\mathcal{D}$ leaving log L unchanged.

E-Step: Depends on the parametric expression of $Q_\lambda(Y|X)$.

M-Step: Depends on the parametric expression of $P_\theta(X,Y)$.

This works when the missing value patterns are sufficiently random!

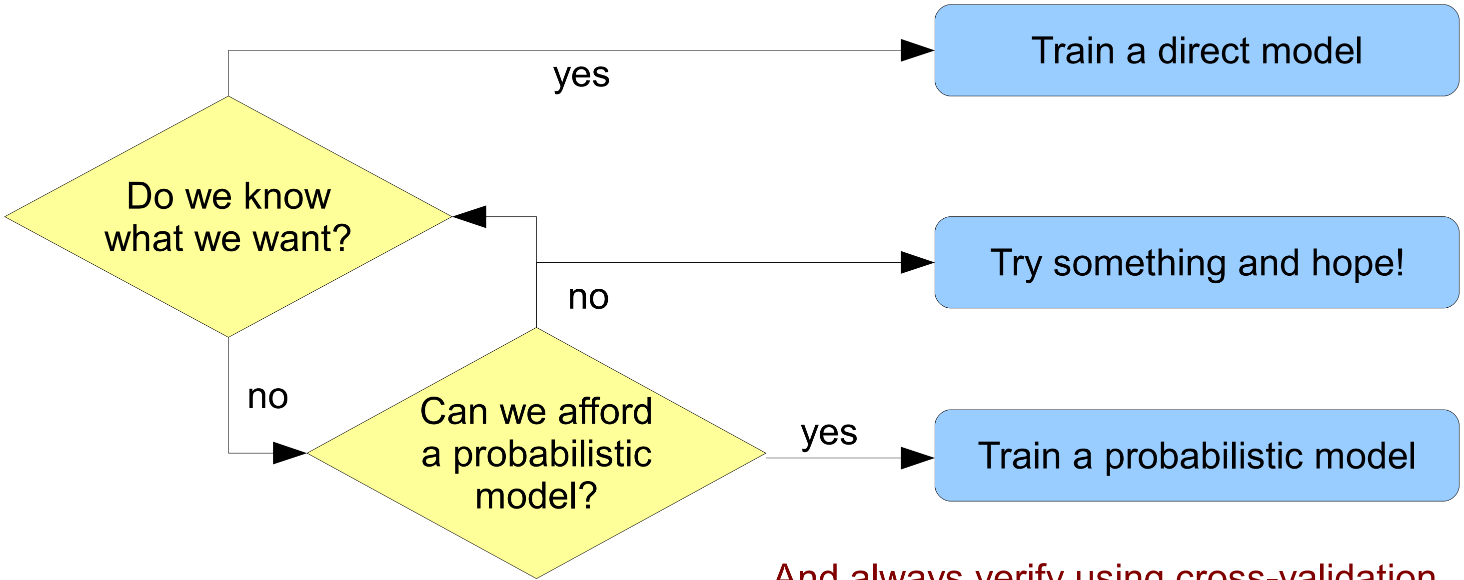
Conclusion

Expectation Maximization

– EM is a very useful algorithm for probabilistic models.
– EM is an alternative to sophisticated optimization
– EM is simpler to implement.

Probabilistic Models

– More versatile than direct approaches.
– More demanding than direct approaches (assumptions, data, etc.)



And always verify using cross-validation.