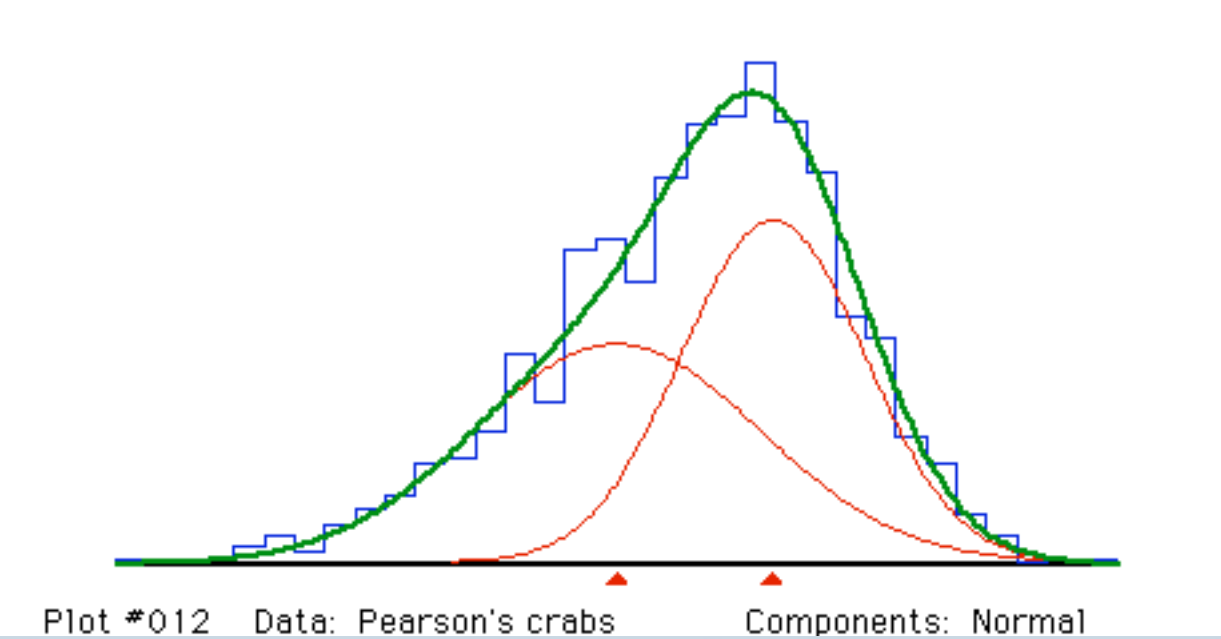
Mixture Models

Data Mining and Statistical Learning
Group Discussion

Napels Crab

- Pearson (1894) studied the ratio of "forehead" breadth to body length for 1000 crabs sampled at Naples by Professor W.F.R. Weldon



Mixture Distribution

- Assumes data is an *i.i.d* sample from some population described by a mixture distribution

$$p(x) = \sum_{k=1}^{K} \pi_k p_k(x; \theta_k)$$

- A Mixture Gaussian distribution is commonly used

$$p(x) = \sum_{k=1}^{K} \pi_k N(\mu_k, \Sigma_k)$$

Mixture Distribution

- Another way to understand the mixture distribution:

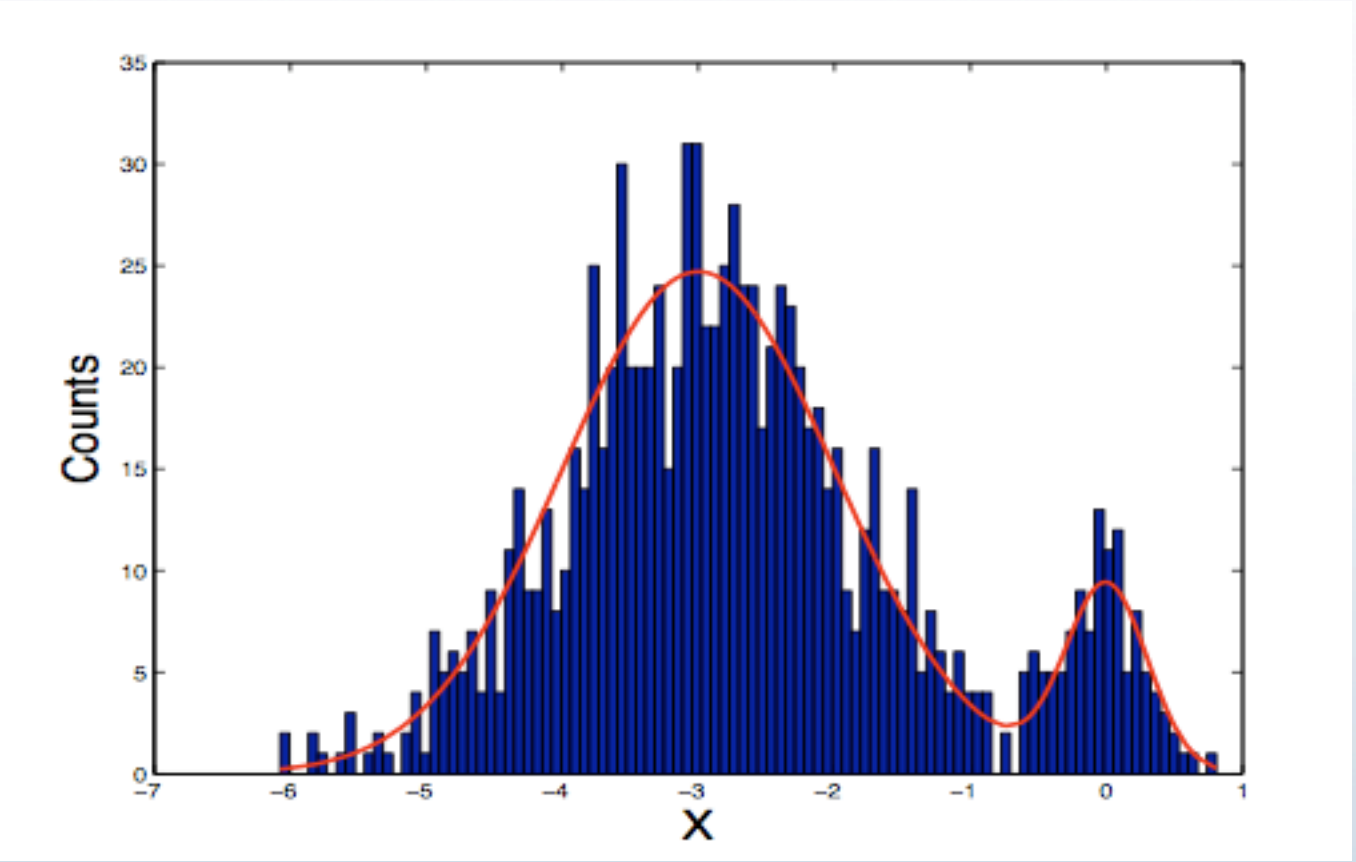
$$Z_k \sim p_k(\cdot, \theta_k)$$

$$(I_1, \cdots, I_K) \sim multinomial(1, \pi_1, \cdots, \pi_K)$$

$$X = \sum_{k=1}^{K} I_k Z_k$$

- ***X*** follows a mixture distribution $p(x) = \sum_{k=1}^{K} \pi_k p_k(x; \theta_k)$

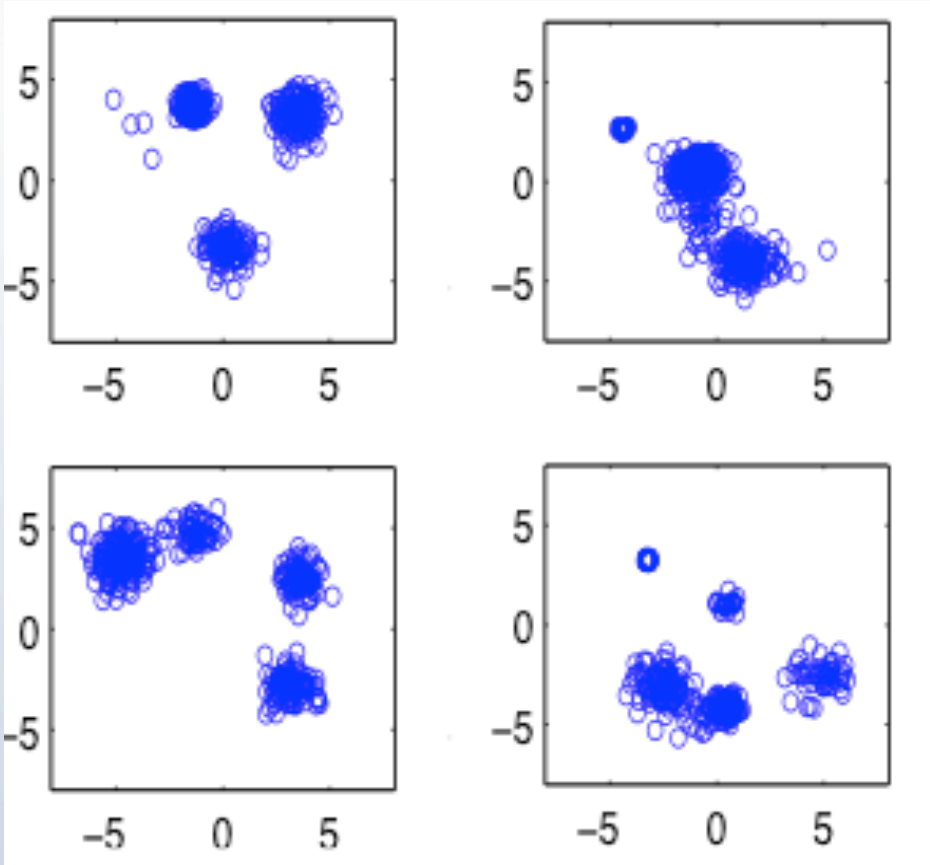
One Example



$$p(x) = 0.9N(-3, 1^2) + 0.1N(0, 0.3^2)$$

More Examples

- Mixture of six Gaussians
- Number of points from each Gaussian:
  - Multinormail (400, 1/6, …, 1/6)
- Mean and SD of each Gaussian randomly sampled from
  - unif([-5, 5] x [-5, 5]) and
  - unif(0,0.8)



Use of Mixture Model

- Parametric Density Estimation

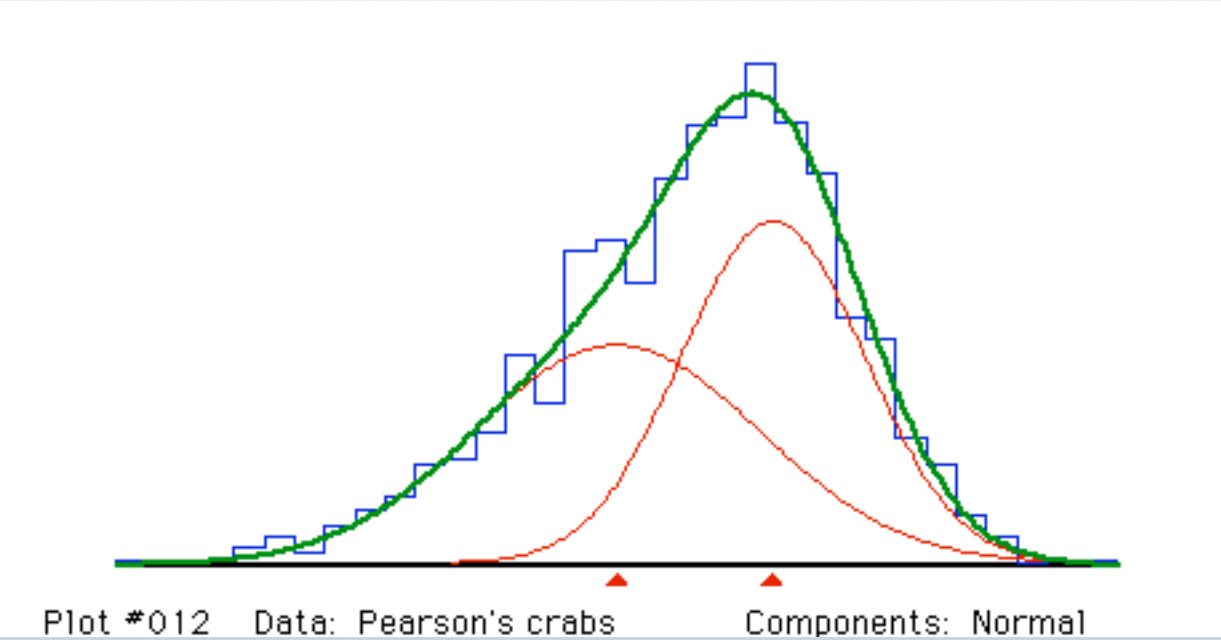
$$\hat{p}(x) = \sum_{k=1}^{K} \hat{\pi}_k p_k(x; \hat{\theta}_k)$$

- Nonparametric (kernel) Density Estimation
- Classification or Clustering

$$\max_{k=1,\cdots,K} P(I_k = 1|x)$$

Use of Mixture Model

- Density Estimation
- Classification or Clustering



Application of Mixture Model

- Astronomy
- Genetics
- Medical Science
- Computer Vision
- Speech Recognition
- …

Parameter Estimation

- Method of Moment
- Maximum Likelihood Estimation
- Bayesian inference of the posterior distribution

Method of Moment

- Pearson (1894) used MoM to estimate a mixture Gaussion distribution with two components on the Napels Crab data
- Express the moments $M^t(\pi, \theta) = E(X^t)$ of *p(x)* in terms of the parameters (π, θ) by theoretically calculation
- Compute the empirical moments $\overline{M}^t = (\sum_{i=1}^{n} X_i^t)/n$ from observation
- Find estimate $(\hat{\pi}, \hat{\theta})$ by solving $M^t(\pi, \theta) = \overline{M}^t$

MoM for Mixture Models

- Consider the simplest Mixture Distribution: Mixture of two univariate Gaussians
  - Five parameters: $(\pi_1, \mu_1, \sigma_1, \mu_2, \sigma_2)$
  - One may match the first five moments and solve for the parameters
  - The calculation can be converted as computing a suitable root of a ninth-degree polynomial
  - The solution is not guaranteed to exist or to be unique
- Expression for estimates of three univariate Gaussians was obtained in 50's, but very hard to compute
- Infeasible for large dimension or # of components

Maximum Likelihood Estimate

- MLE is very commonly used technique in fitting mixture models
- Given the density function

$$p(x) = \sum_{k=1}^{K} \pi_k p_k(x; \theta_k)$$

and observations x_1, ..., x_n, the likelihood is:

$$L(\pi, \theta) = \Pi_{i=1}^{n} p_{\pi,\theta}(x)$$

- Optimization routine is usually needed to find the solution that maximizes the likelihood function

Optimization

- In general, an optimization problem tries to find the maximum of a function *f(a)*, in this case the likelihood function
  - Grid Searching
  - Bisection
  - Coordinate Ascent
  - Newton-Raphson
  - A lot more ……

Expectation/Maximum Algorithm

- EM algorithm is tied to models with incomplete observations
- Suppose the ideal observations $T \sim P_\alpha$ with density $p_\alpha(t)$
- However, we observed $S \equiv S(T) \sim Q_\alpha$ with density $q_\alpha(s)$
- ***S*** represents part of ***T***, and the rest of ***T*** is "missing" and its "reconstruction" is part of the process of estimating α by maximum likelihood

Mixture Distribution

- Another way to understand the mixture distribution:

$$Z_k \sim p_k(\cdot, \theta_k)$$

$$(I_1, \cdots, I_K) \sim multinomial(1, \pi_1, \cdots, \pi_K)$$

$$X = \sum_{k=1}^{K} I_k Z_k$$

- In this case,

$$T = (I_1, \cdots, I_K, Z_1, \cdots, Z_K) \qquad S = X$$

$$\alpha = (\pi_1, \cdots, \pi_K, \theta_1, \cdots, \theta_K)$$

How does EM work?

- Define

$$J(\alpha|\alpha_0) \equiv E_{\alpha_0}\left(\log \frac{p(T,\alpha)}{p(T,\alpha_0)} \mid S(T) = s\right)$$

- Initialize with $\alpha_{old} = \alpha_0$
- E-step: compute $J(\alpha|\alpha_0)$ for as many values of α as needed. If this is difficult, the EM algorithm is probably not suitable
- M-step: maximize $J(\alpha|\alpha_0)$ as a function of α. Again, if difficult, EM may not be appropriate
- Set $\alpha_{new} = \arg\max J(\alpha|\alpha_{old})$, reset $\alpha_{old} = \alpha_{new}$ and repeat the E and M steps until converge

An Example

- Consider the simplest Mixture Distribution: Mixture of two univariate Gaussians
  - Five parameters: $\alpha = (\pi_1, \mu_1, \sigma_1, \mu_2, \sigma_2)$
- Initialization: $\hat{\alpha} = (\hat{\pi}_1, \hat{\mu}_1, \hat{\sigma}_1, \hat{\mu}_2, \hat{\sigma}_2)$
- E-step: compute the *responsibilities*

$$\hat{\gamma}_i = \frac{\hat{\pi}_1 \phi_{\hat{\mu}_1, \hat{\sigma}_1}(x_i)}{\hat{\pi}_1 \phi_{\hat{\mu}_1, \hat{\sigma}_1}(x_i) + (1 - \hat{\pi}_1)\phi_{\hat{\mu}_2, \hat{\sigma}_2}(x_i)}$$

which is a estimate of

$$\gamma_i = E(I_{i,1}|\alpha, X_i) = P(I_{i,1} = 1|\alpha, X_i)$$

An Example

- M-step: compute the weighted means and variances:

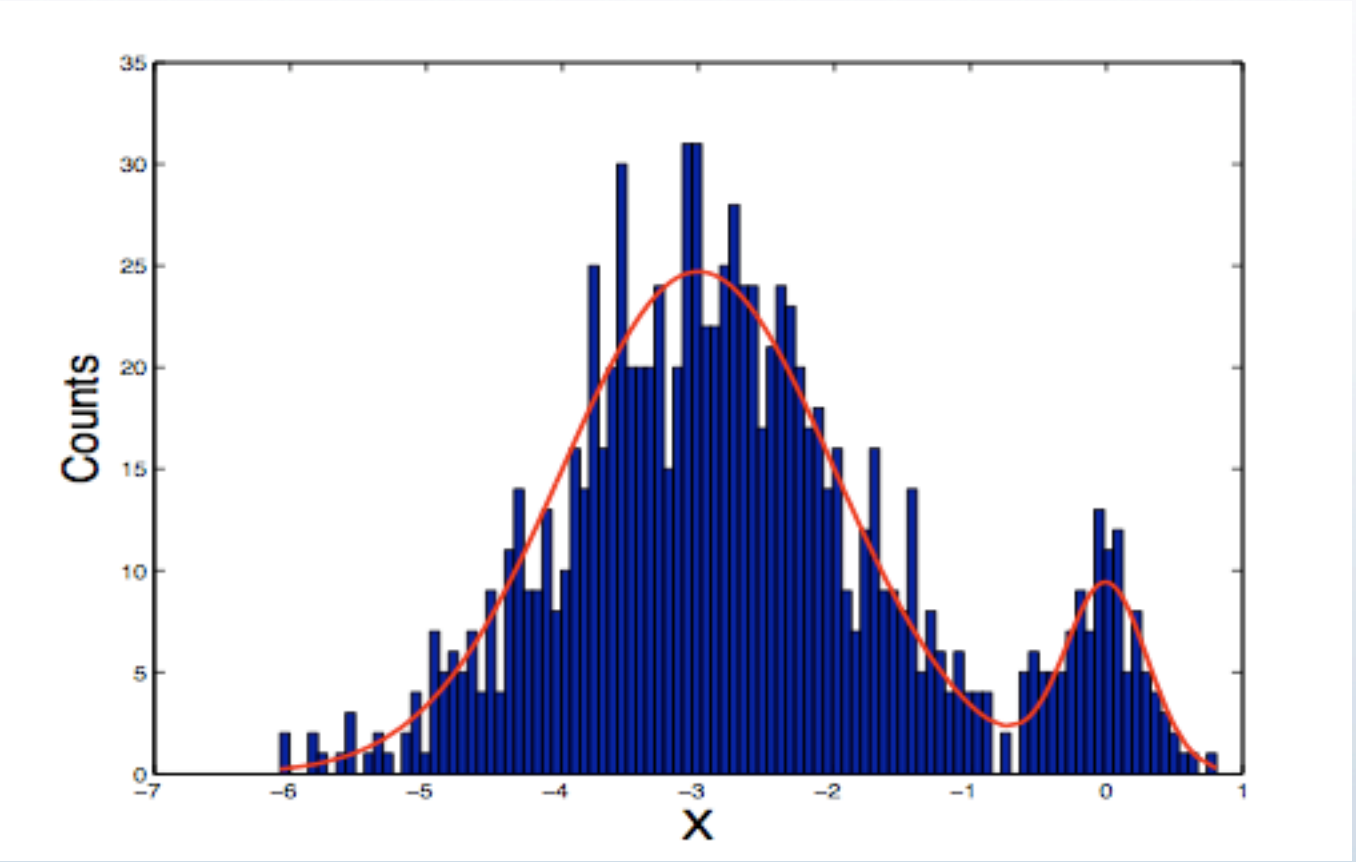
$$\hat{\mu}_1 = \frac{\sum_{i=1}^n \hat{\gamma}_i x_i}{\sum_{i=1}^n \hat{\gamma}_i} \qquad \hat{\sigma}_1^2 = \frac{\sum_{i=1}^n \hat{\gamma}_i (x_i - \hat{\mu}_1)^2}{\sum_{i=1}^n \hat{\gamma}_i}$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^n (1-\hat{\gamma}_i) x_i}{\sum_{i=1}^n (1-\hat{\gamma}_i)} \qquad \hat{\sigma}_2^2 = \frac{\sum_{i=1}^n (1-\hat{\gamma}_i)(x_i - \hat{\mu}_2)^2}{\sum_{i=1}^n (1-\hat{\gamma}_i)}$$

and the mixing probability $\hat{\pi}_1 = \sum_{i=1}^n \hat{\gamma}_i / n$

- Iterate the E-step and M-step until convergence

Example Revisited



$$p(x) = 0.9N(-3, 1^2) + 0.1N(0, 0.3^2)$$

Initialization

- EM algorithm is a local optimization method that is guaranteed to converge to a local maximum
- Since the likelihood functions of most mixture models are not a concave function, the local maximum found by the EM algorithm may not be the global maximum
- Actually, the result from the EM algorithm is highly sensitive to the initialization
- How to do a good job in initialization?
  - try multiple random starting location
  - use results from other fast method, e.g. Kmeans

Combinational Algorithms

- Each point x_i is assigned to one of ***K*** clusters through a many-to-one mapping $k = C(i)$
- The quality of the clustering assignment C is measured by a loss function

$$W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(j)=k} d(x_i, x_j)$$

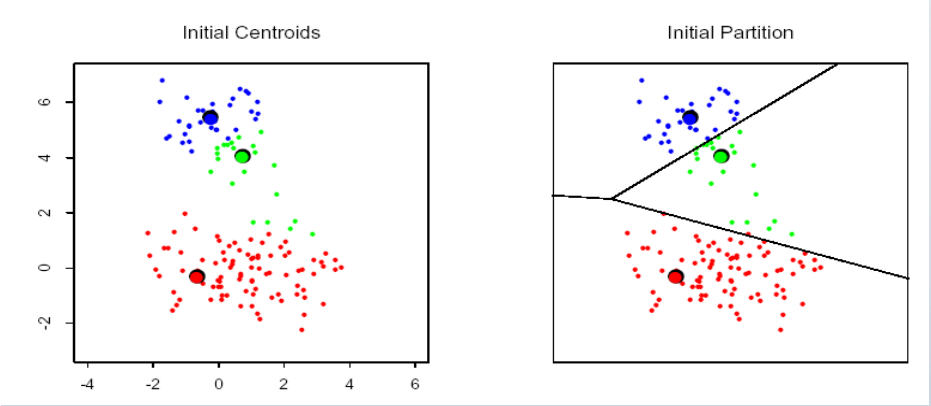
- Direct optimization of $W(C)$ is computationally infeasible even for datasets of ordinary size
- K-means algorithm is one of the most popular iterative descent method try to find the minimum of $W(C)$

K-means Algorithms

- Given number of clusters ***K*** and an initial set of ***K*** seed points (serve as the centroid of clusters)
  1. Partition the items into ***K*** initial clusters by assigning each point to its closest center
  2. Recalculate the ***mean*** for each cluster, use it to replace the centriod that is use to generate the current cluster
  - repeat steps 1 and 2 until convergence
- One may replace the ***mean*** by ***median*** in step 2 to get K-median algorithm

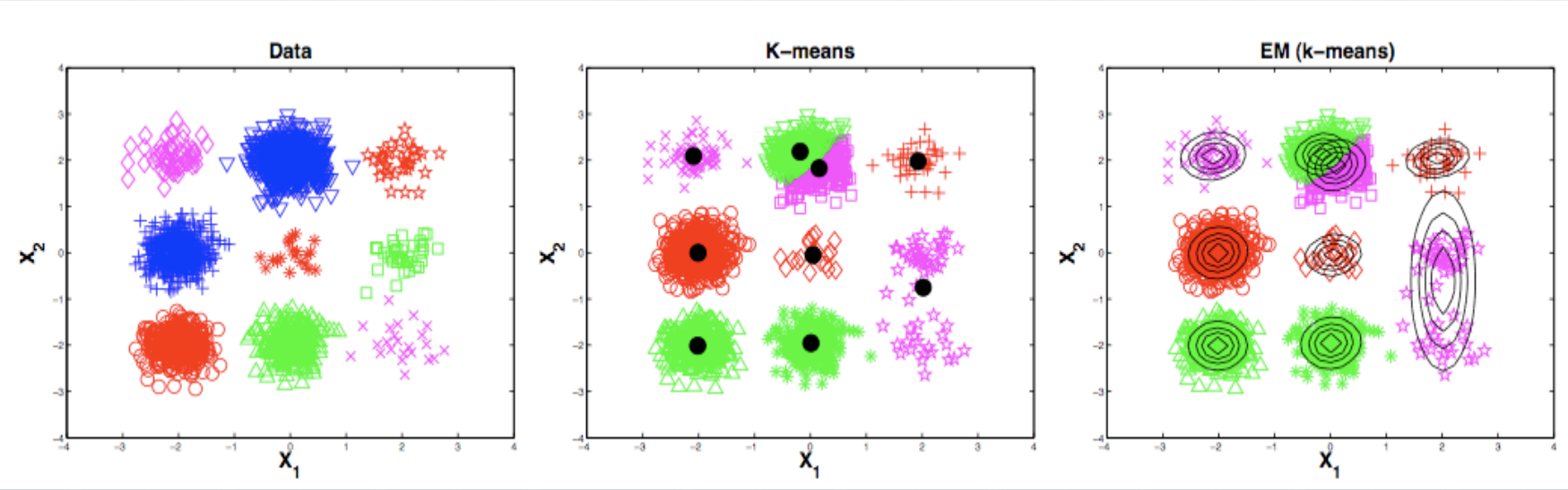
K-means Algorithms

- Something you should know about K-means:
  - Converges very fast
  - User need to provide the number of groups
  - A local optimization routine, sensitive to initial values
  - Should be started with many different random choices of initial values



EM with Kmeans initialization

- EM is also a local optimization routine, which is sensitive to initialization
- In practice, EM is usually initialized with the results from a K-means algorithm



Napels Crab

- A single Weibull component is also an acceptable fit