- k means doesn't work well
  - if clusters are overlapping (soft-weights help)
  - non-circular clusters

- what if we think about data having been created by some collection of probability models
  - ↳ want to determine models (parameters from data points
  - ↳ don't know which points came from which model

know models → easy to figure out which model produced point

know which points belong to a model

↳ easy to determine model

↳ we did this earlier in semester

sounds similar to k-means

Iterative Algorithm: Expectation-Maximization (EM)

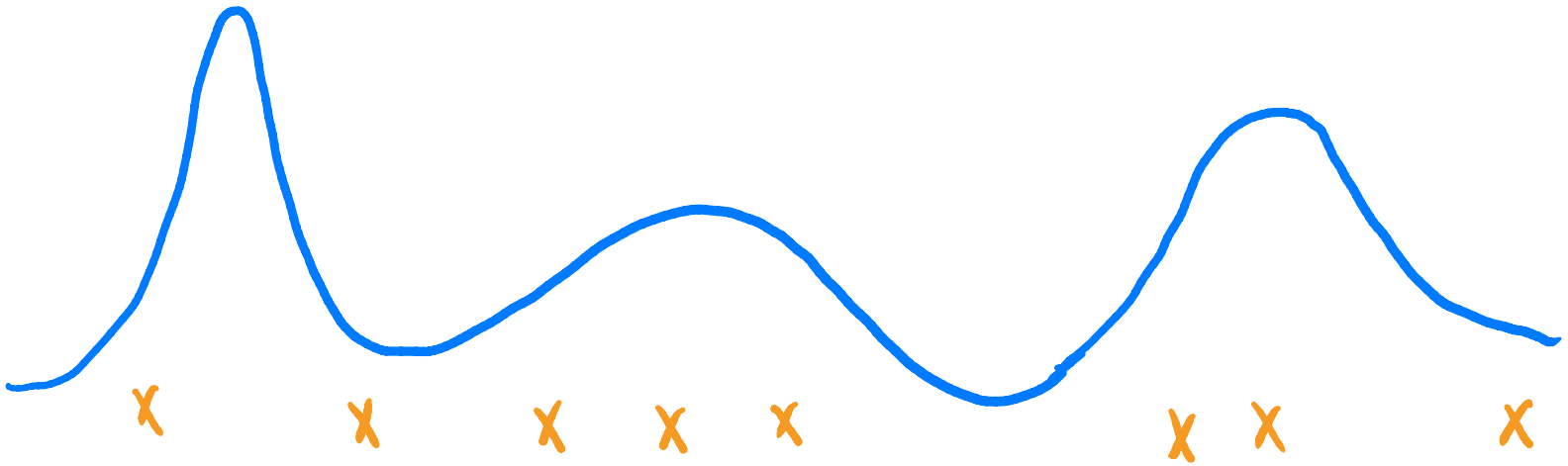
iterating through 2 steps

- updating which points belong to which model (use weights since models might overlap)
- updating parameters of models

GMM

- each cluster modeled as Gaussian
  - ↳ model with mean and covariance

Call data modeled as mixture of Gaussian



↑ 1D, typically have multivariate

- want to find parameters that maximize likelihood of data

  model parameters

  mean, variance, gaussian size

  → use EM

EM for GMM

- iterate through 2 steps until convergence (parameters stop changing)
- each step increases log-likelihood of model

Expectation Step

have parameter values → update weights

Compute weights for each point

$$w_{ic} = \frac{\pi_c \mathcal{N}(x_i : \mu_c, \Sigma_c)}{\sum_k \pi_k \mathcal{N}(x_i : \mu_k, \Sigma_k)}$$

probability x_i belongs to model c

Maximization Step

have weights → update parameters

k: # clusters

$$\pi_c = \frac{\sum_i w_{ic}}{k}$$

$$\mu_c = \frac{\sum w_{ic}\, x_i}{\sum_i w_{ic}}$$

$$\sum_c = \frac{\sum_i w_{ic} (x_i - \mu_c)^T (x_i - \mu_c)}{\sum_i w_{ic}}$$

EM:

algorithm is more general

GMM just one example

topic models are another

exact formula for update is different for each

depends on maximizing log-likelihood using latent variables