Lab07 - Advanced clustering (GMM, DBSCAN, JP)

1) Maximum Likelihood

Example taken from
Wikipedia: https://en.wikipedia.org/wiki/Maximum_likelihood#Discrete_distribution.2C_finite_parameter_space

Suppose one wishes to determine just how biased an unfair coin is. Call the probability of tossing a HEAD p. The goal then becomes to determine p.

Suppose the coin is tossed 80 times: i.e., the sample might be something like x_1 = H, x_2 = T, ..., x_{80} = T, and the count of the number of HEADS "H" is observed.

The probability of tossing TAILS is 1 – p (so here p is θ above). Suppose the outcome is 49 HEADS and 31 TAILS, and suppose the coin was taken from a box containing three coins: one which gives HEADS with probability p = 1/3, one which gives HEADS with probability p = 1/2 and another which gives HEADS with probability p = 2/3. The coins have lost their labels, so which one it was is unknown. Using **maximum likelihood estimation** the coin that has the largest likelihood can be found, given the data that were observed. By using the probability mass function of the binomial distribution with sample size equal to 80, number successes equal to 49 but different values of p (the "probability of success"), the likelihood function (defined below) takes one of three values:

$$\Pr(\mathrm{H}=49 \mid p=1/3)=\binom{80}{49}(1/3)^{49}(1-1/3)^{31}\approx 0.000,$$

$$\Pr(\mathrm{H}=49 \mid p=1/2)=\binom{80}{49}(1/2)^{49}(1-1/2)^{31}\approx 0.012,$$

$$\Pr(\mathrm{H}=49 \mid p=2/3)=\binom{80}{49}(2/3)^{49}(1-2/3)^{31}\approx 0.054.$$

The likelihood is maximized when p = 2/3, and so this is the *maximum likelihood estimate* for p.

The probability of getting exactly k successes in n trials is given by the probability mass function:

$$\Pr(X=k)=\binom{n}{k}p^k(1-p)^{n-k}$$

(formula for matlab/octave: nchoosek(n,k)*(p)^k*(1-p)^(n-k))
where

$$\binom{n}{k}=\frac{n!}{k!(n-k)!}$$

is the *binomial coefficient*: we want exactly k successes and n-k failures; the k successes can occur anywhere among the n trials, and the binomial coefficient measures all the different ways we can get them.

In R:

n = 80
k = 49
p = 1/3
choose(n,k)*(p)^k*(1-p)^(n-k)

p = 1/2
choose(n,k)*(p)^k*(1-p)^(n-k)

p = 2/3
choose(n,k)*(p)^k*(1-p)^(n-k)

2) Related material

- Gaussian Mixture models material by prof. Andrew W. Moore
- Gaussian Mixtures demo
- Local link to DBSCAN paper here
- Local link to Jarvis-Patrick paper here