COMPSCI 650 Applied Information Theory 4/21/2016

Lecture

Instructor: Arya Mazumdar *Scribe:* [redacted]

1 Clustering

We want to look at $x_1, x_2, x_3, ..., x_n$ samples and assign them to 1 *to* k clusters. k can be known or unknown.

Previously we had this problem in hypothesis testing that we pick up a sample and decide whether it is H_1 or H_2.

We have probability distribution of hypothesis H_1 and H_2 as follow

$$H_1 : f_1(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}} \tag{1}$$

$$H_2 : f_2(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_2)^2}{2\sigma^2}} \tag{2}$$

Prior probability of H_1 and H_2 is P_1 and P_2 respectively. So we have:

$$f(x) = P_1 f_1(x) + P_2 f_2(x) = \sum_{k=1}^{2} P_k f_k(x) \Rightarrow \tag{3}$$

$$f(x_n) = \sum_{k=1}^{2} P_k \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_n-\mu_k)^2}{2\sigma^2}} \qquad 1 \le n \le N \tag{4}$$

The above equation is called a Gaussian mixture model.

Assigning labels is an inference problem. Here we want to figure out μ_k. However, in hypothesis testing μ_k is known.

Since samples $x_1, x_2, ..., x_N$ are i.i.d we can write:

$$f(x_1, x_2, ..., x_n) = \prod_{n=1}^{N} f(x_n) \tag{5}$$

$$\log f(x_1, x_2, ..., x_N) = \sum_{n=1}^{N} \log f(x_n) = L \tag{6}$$

Given an observation we want to find μ_k in a way that function $f(x_1, x_2, ..., x_n)$ will be maximized. So we should find $\frac{\partial L}{\partial \mu_1}$ and $\frac{\partial L}{\partial \mu_2}$.

First we calculate the probability that a sample comes from hypothesis H_1:

$$P(k_n = 1|x_n) = f(x_n|k_n = 1)\frac{P(k_n = 1)}{f(x_n)} \xrightarrow{We\ assume\ P_1=P_2=1/2} \tag{7}$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu_1)^2}{2\sigma^2}} \frac{1/2}{\frac{1}{2}\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu_1)^2}{2\sigma^2} + \frac{-(x-\mu_2)^2}{2\sigma^2}}} \tag{8}$$

$$= \frac{1}{1 + e^{\frac{1}{2\sigma^2}((x-\mu_1)^2 - (x-\mu_2)^2)}} \tag{9}$$

$$P_{1|n} = \frac{1}{1 + e^{\frac{1}{2\sigma^2}\left(2(\mu_2-\mu_1)x - (\mu_1^2 - \mu_2^2)\right)}} \tag{10}$$

$$P_{2|n} = \frac{1}{1 + e^{-\frac{1}{2\sigma^2}\left(2(\mu_2-\mu_1)x - (\mu_1^2 - \mu_2^2)\right)}} \tag{11}$$

Now we calculate $\frac{\partial L}{\partial \mu_1}$:

$$\frac{\partial L}{\partial \mu_1} = \sum_{n=1}^{N} \frac{\partial}{\partial \mu_1} \log f(x_n) \tag{12}$$

$$\log f(x_n) = -\log 2 - \frac{1}{2}\log(2\pi\sigma^2) + \log\left(e^{\frac{(x-\mu_1)^2}{2\sigma^2}} + e^{\frac{(x-\mu_2)^2}{2\sigma^2}}\right) \tag{13}$$

$$(12), (13) \Rightarrow \frac{\partial log f(x_n)}{\partial \mu_1} = \frac{\left(e^{-\frac{(x-\mu_1)^2}{2\sigma^2}}\right) \times \left(\frac{2(x-\mu_1)}{2\sigma^2}\right)}{e^{-\frac{(x-\mu_1)^2}{2\sigma^2}} + e^{-\frac{(x-\mu_2)^2}{2\sigma^2}}} \tag{14}$$

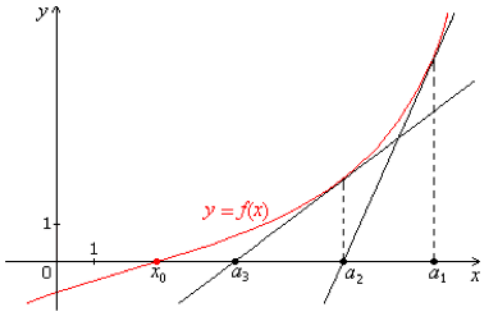
$$= P_{1|n}\frac{x-\mu_1}{\sigma^2} \tag{15}$$

So we have:

$$\frac{\partial L}{\partial \mu_1} = \sum_{n=1}^{N} P_{1|n}\frac{x-\mu_1}{\sigma^2}$$

$$\frac{\partial L}{\partial \mu_2} = \sum_{n=1}^{N} P_{2|n}\frac{x-\mu_2}{\sigma^2}$$

To find μ_k we use Newton's method described below.
Newton's Method:



$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

So we have:

$$\mu_1^{(i+1)} = \mu_1^{(i)} - \frac{\frac{\partial L}{\partial \mu_1^{(i)}}}{\frac{\partial^2 L}{\partial (\mu_1^{(i)})^2}} \tag{16}$$

$$\frac{\partial^2 L}{\partial (\mu_1)^2} = \sum_{n=1}^{N} \left(P_{1|n}(-\frac{1}{\sigma^2}) + \frac{x-\mu_1}{\sigma^2} \frac{\partial P_{1|n}}{\partial \mu_1} \right) \tag{17}$$

$$\approx -\sum_{n=1}^{N} P_{1|n} \frac{1}{\sigma^2} \tag{18}$$

$$(16), (18) \Rightarrow \mu_1^{(i+1)} = \mu_1^{(i)} + \frac{\sum_{n=1}^{N} P_{1|n} \frac{x-\mu_1^{(i)}}{\sigma^2}}{\sum_{n=1}^{N} P_{1|n} \frac{1}{\sigma^2}} \tag{19}$$

$$\mu_1^{(i+1)} = \mu_1^{(i)} + \frac{\sum_{n=1}^{N} P_{1|n} x_n}{\sum_{n=1}^{N} P_{1|n}} - \mu_1^{(i)} \tag{20}$$

$$= \frac{\sum_{n=1}^{N} P_{1|n} x_n}{\sum_{n=1}^{N} P_{1|n}}, \tag{21}$$

$$\mu_2^{(i+1)} = \frac{\sum_{n=1}^{N} P_{2|n} x_n}{\sum_{n=1}^{N} P_{2|n}} \tag{22}$$

1.1 K-means Algorithm

To cluster N samples to k clusters, we do as follow:

- First iteration: Choose k arbitrary points $\mu_1^{(0)}, \mu_2^{(0)}, ..., \mu_k^{(0)}$, then assign other points to their nearest $\mu_1^{(0)}, \mu_2^{(0)}, ..., \mu_k^{(0)}$
- $i = 1$
- Then for each iteration i:
  - compute points' center of mass $(\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_k^{(i)})$.
  - check if means$(\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_k^{(i)})$ converged, return
  - assign points to the nearest center of mass $\mu_1^{(i)}, \mu_2^{(i)}, ..., \mu_k^{(i)}$
  - $i = i + 1$

An extension of k-means algorithms is soft k-means which instead of labelling we membership(probability to be a member of a cluster).