Homework 2

Chujie Chen
cchen641@gatech.edu

October 17, 2019

1 EM for Mixture of Gaussians

(a)

From (2)

$$\begin{aligned}
p(x) &= \sum_{z\in Z} p(z)p(x|z) \\
&= \sum_{z\in Z} \Pi_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma_k)^{z_k} \\
&= \sum_{z\in Z} \Pi_{k=1}^{K} [\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)]^{z_k}
\end{aligned}$$

For each $z = z^{(i)} = [0, ..., 1, ..., 0]$. (only the i-th element is nonzero)

$$\Pi_{k=1}^{K} [\pi_k \mathcal{N}]^{z_k} = \pi_i \mathcal{N}(x|\mu_i, \Sigma_i)$$

Thus,

$$\begin{aligned}
p(x) &= \sum_{z\in Z} \pi_i \mathcal{N}(x|\mu_i, \Sigma_i) \\
&= \pi_1 \mathcal{N}(x|\mu_1, \Sigma_1) + ... + \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \\
&= \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) = \boxed{(1)}
\end{aligned}$$

(b)

$$\begin{aligned}
p(z_k^n|x_n) &= \frac{p(x_n|z_k^n)p(z_k^n)}{p(x_n)} \\
&= \frac{\mathcal{N}(x_n|\mu_k, \Sigma_k)^{z_k^n} \pi_k^{z_k^n}}{\sum_j^K \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)}
\end{aligned}$$

More specifically, the responsibilities are

$$\begin{aligned}
\tau_k^n &= p(z_k^n = 1|x_n) + 0 \\
&= \boxed{\frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_j^K \pi_j \mathcal{N}(x_n|\mu_j, \Sigma_j)}}
\end{aligned}$$

(c)

To start with derivatives:

$$\begin{aligned}
\frac{\partial lnp(X|\pi, \mu, \Sigma)}{\partial \mu_k} &= \frac{\partial}{\partial \mu_k} \sum_n ln \sum_k \pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k) \\
&= -\sum_n \frac{\pi_k \mathcal{N}(x_n|\mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(x_n|mu_j, \Sigma_j} \Sigma_k^{-1}(x_n - \mu_k) = 0
\end{aligned}$$

We will have

$$\boxed{\mu_k = \frac{\sum_n \gamma(z_{nk})x_n}{\sum_n \gamma(z_{nk})} = \frac{\sum_n \gamma(z_{nk})x_n}{N_k}}$$

Similarly,

$$\frac{\partial lnp}{\partial \Sigma_k} = \sum_n \frac{\pi_k \mathcal{N}}{\sum_j \pi_j \mathcal{N}}[-\frac{1}{2}\Sigma^{-1} + \frac{1}{2}(x-\mu_k)^T\Sigma^{-2}(x-\mu_k)] = 0$$

We have

$$\boxed{\Sigma_k = \frac{\sum_n \gamma(x-\mu_k)(x-\mu_k)^T}{\sum_n \gamma} = \frac{\sum_n \gamma(z_{nk})(x-\mu_k)(x-\mu_k)^T}{N_k}}$$

Lastly, derivative with constrains

$$\frac{\partial}{\partial \pi_k}[lnP + \lambda(\sum_j \pi_j - 1)] = \sum_n \frac{\mathcal{N}}{\sum_j \pi_j \mathcal{N}} + \lambda = 0$$

We get

$$\lambda = -\sum_n \frac{\mathcal{N}}{\sum_j \pi_j \mathcal{N}}$$

To sum over both sides with π_j, we have

$$\sum_j \lambda\pi_j = \lambda = -\sum_n \sum_i \frac{\mathcal{N}}{\sum_j \pi_j \mathcal{N}} = -N$$

Thus, we have

$$0 = \sum_n \frac{\mathcal{N}}{\sum_j \pi_j \mathcal{N}} - N$$

$$N\pi_k = \sum_n \frac{_k\mathcal{N}}{\sum_j \pi_j \mathcal{N}}$$

Thus,

$$\boxed{\pi_k = \frac{\sum_n \gamma}{N} = \frac{N_k}{N}}$$

(d)

We take a look at resposibility

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)}{\sum_j \pi_j \mathcal{N}(x|\mu_j, \Sigma_j)}$$
$$= \frac{\pi_k exp[-||x_n - \mu_k||^2/2\epsilon]}{\sum_j \pi_j exp[-||x_n - \mu_k||^2/2\epsilon]}$$

When $\epsilon \to 0$

$$\gamma(z_{nk}) = 0 \quad (for \quad k \neq j)$$
$$\gamma(z_{nk}) = 1 \quad (otherwise)$$

Thus, it becomes hard assignments

$$\gamma(z_{nk}) = r_{nk}$$

In the meantime, the update formula becomes

$$\pi_k = \frac{1}{N}\sum_n r_{nk}$$

$$\Sigma_k = \frac{1}{\sum_n r_{nk}} \sum_n r_{nk}(x_n - \mu_k)(x_n - \mu_k)^T$$

From (a), we have the expected value of the complete-data log likelihood function:

$$\begin{aligned}
\mathbb{E}[lnp(X, Z|\mu, \Sigma, \pi)] &= \sum_n \sum_k \gamma(z_{nk})[ln\pi_k + ln\mathcal{N}(x_n|\mu_k, \Sigma_k)] \\
&= -\frac{1}{2} \sum_n \sum_k r_{nk}||x_n - \mu_k||^2 + const.
\end{aligned}$$

To compare this with objective function

$$J = \sum_n \sum_k \gamma_{nk}||x_n - \mu_k||^2$$

We can see that maximizing the expected complete data log-likelihod for this medel is equivalent to minimizing above objective function in K-means.

2 Density Estimation

(a)

$$\begin{aligned}
\mathcal{L} &= log\Pi_i(h_i)^{n_i} \\
&= \boxed{\sum_i n_i log h_i}
\end{aligned}$$

(b)

We can write log likelihood function with constraint as below

$$\mathcal{L}' = \mathcal{L} + \lambda(\sum_i h_i \Delta_i - 1)$$

$$\frac{\partial \mathcal{L}'}{\partial h_j} = \frac{n_j}{h_j} + \lambda \Delta_j = 0$$

We have

$$\sum_j n_j = N = -\lambda \sum_j h_j \Delta_j = -\lambda$$

Thus,

$$\hat{h}_j = \frac{n_j}{-\lambda \Delta_j} = \boxed{\frac{n_j}{N \Delta_j}}$$

(c)

(1)

F. One can assume they have many parameters so they don't have a specific shape. Non-parameter means they cannot be descibed with fixed number of parameters.

(2)

F. It depends on the dataset itself. For example, if the data were purely generated by a gaussian, then a gaussian kernel would be the best option for that dataset.

(3)

F. High dimensinal data requires n^d bins. Besides, if the number of bins is greater than the number of data, many bins will be empty.

(4)

T. With fixed number of parameter, we have some "shape" of the pdf.

3 Information Theory

(a)

We prove the chain rule first,

$$\begin{aligned}
H(X,Y) &= -\sum_x \sum_y p(X,Y) logp(X,Y) \\
&= -\sum_x \sum_y p(x,y) log[p(Y|X)p(X)] \\
&= -\sum_x \sum_y p(x,y) logp(y|x) - \sum_x \sum_y p(x,y) logp(x) \\
&= H(Y|X) - \sum_x p(x) logp(x) \\
&= H(Y|X) + H(X)
\end{aligned}$$

And since

$$0 \leq I(X,Y) = H(X) - H(X|Y)$$

We have

$$H(X|Y) \leq H(Y)$$

Thus,

$$\boxed{H(X,Y) \leq H(X) + H(Y)}$$

With the quality if and only if x and y are independent.

(b)

From chain rule,

$$H(Y|X) = H(X,Y) - H(X)$$

We have

$$\boxed{I(X,Y) = H(Y) - H(Y|X) = H(X) + H(Y) - H(X,Y)}$$

(c)

From the chain rule,

$$H(X,Z) = H(Z) + H(X|Z) = H(X) + H(Z|X)$$

So we have

$$\begin{aligned}
H(Z) &= H(X) + H(Z|X) - H(X|Z) \\
&= H(X) + H(Y|X) - H(X|Y) \\
&= H(Y) + H(X|Y) - H(Y|X)
\end{aligned}$$

We can discuss here: if X, Y are independent, then $H(Y|X) = H(Y)$, then we have

$$H(Z) = H(X) + H(Y) - H(X|Z)$$

This requires $H(X|Z) = 0$, which means$z = x + y$ is unique ($x_1 + y_1 \neq x_2 + y_2$) for any pairs of X and Y.

4 Programming: Image compression

4.1 EM for Mixture of Multinomials

The converge condition was set as the μ and π start to converge:

```
while(norm(mu - old_mu) > 10^(-9) || norm(pi - old_pi) > 10^(-9))
    E-step
    M-step
end
```

The runtime and accuracies from 10 runs on the toy dataset are listed below:

#	Runtime (s)	Accuracy (%)
1	7.1617	78
2	1.6955	82.25
3	5.0033	70
4	2.5036	71.5
5	5.0723	66
6	1.8409	79.25
7	4.7195	82.25
8	3.1392	85.75
9	2.0552	82.75
10	2.7338	74.75
Average	3.5915	77.25

4.2 Extra Credit: Realistic Topic Models

The converge condition was set as the $P(w|z)$, $P(d|z)$ and $P(z)$ start to converge:

```
threshold = 10^(-2);
while(norm(pwgz-old_pwgz)>threshold || norm(pdgz-old_pdgz)>threshold
    || norm(pz-old_pz)>threshold)
    E-step
    M-step
end
```

Results are below:

$n\_topics = 2$

Time: 17.9892s
W 1: learning,data,neural,output,network,networks,information,figure,algorithm,set,
W 2: model,network,time,number,input,function,learning,error,figure,set,

$n\_topics = 3$

Time: 34.5225s
W 1: model,learning,neural,function,network,time,input,figure,training,networks,
W 2: network,neural,set,figure,time,case,learning,model,error,output,
W 3: network,data,set,input,system,learning,error,function,units,networks,

$n\_topics = 4$

Time: 47.9690s
W 1: network,figure,input,data,error,number,training,time,output,algorithm,
W 2: model,learning,network,neural,time,networks,training,data,algorithm,state,
W 3: model,set,learning,neural,data,networks,system,input,figure,training,
W 4: function,network,learning,neural,set,input,time,networks,units,data,

$n\_topics = 5$

Time: 69.4620s
W 1: neural,learning,data,function,input,model,figure,units,time,networks,
W 2: network,learning,neural,model,function,data,system,distribution,time,units,
W 3: learning,output,data,figure,network,neural,noise,time,training,patterns,
W 4: model,input,networks,training,figure,algorithm,set,learning,system,data,
W 5: network,function,time,set,output,input,state,error,learning,training,

Conclusion

More specifically, with number of topics equals 2. If we use smaller tolerance and more words, we will have results like below:
Time: 1152s
W 1: model,network,time,figure,input,neural,system,neurons,learning,neuron,output,control,visual,information,cells, state,cell,function,response,image.
W 2: learning,network,training,data,set,function,neural,networks,algorithm,error,model,input,number,problem,hidden, figure,output,results,time,units.

They do have different topics, but the k needs to be chosen very carefullly.