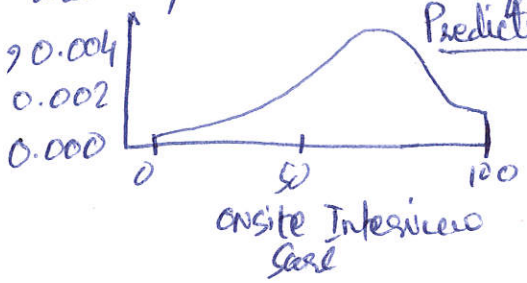
* Latent variable is neither observed during training nor testing. We will use EM algorithm that are used to train latent variable models.

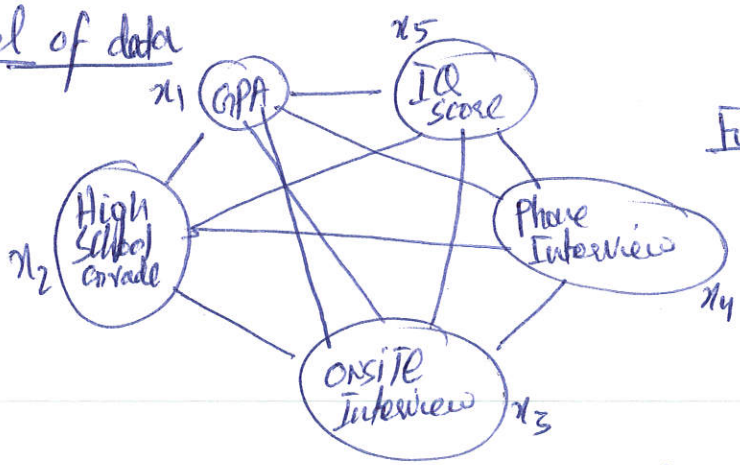
Why do we need probabilistic modelling of DATA?

	High School	GPA	IQ	Phone Interview	Onsite Performance
John	4.0	4.0	120	3/4	?
Helen	3.7	3.6	N/A	4/4	?

We would like to predict onsite Performance. We can use historical data and do regression. But here we have missing variables as well. So, probabilistic modelling of data will help us deal with missing values & it can also provide us confidence over prediction. For instance



Probabilistic Model of data



Fully connected model

The joint distribution table for above model will be ~~very~~ exponentially large. Therefore it is impractical to treat them as parameters.

exponential #s

High school	GPA	IQ	....	Probability
:				

Option#2

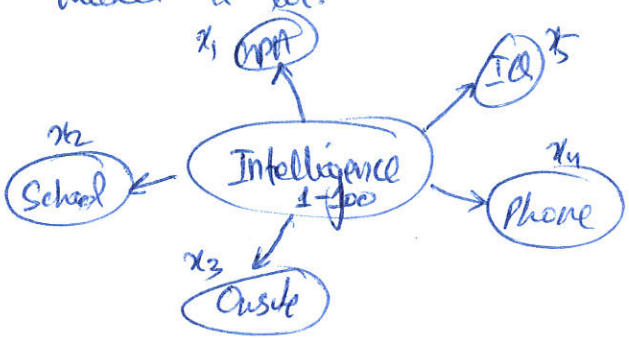
We can ~~build~~ assume parametric model.

$$P(x_1, x_2, x_3, x_4, x_5) = \frac{\exp(-w^T x)}{Z}$$

The problem is that Z is a sum over exponential terms

Option#3

We can introduce latent variable "Intelligence" in the model. It will simplify our model a lot.



$$P(x_1, x_2, x_3, x_4, x_5) = \sum_{I=1}^{100} P(x_1, x_2, x_3, x_4, x_5, I) \quad \text{(Sum rule)}$$

$$= \sum_{I=1}^{100} P(x_1, x_2 \dots x_5 \mid I)\, P(I)$$

$P(x_1, x_2 \dots x_5 \mid I)$: Conditional Prob; $P(I)$: Prior Prob

This conditional prob will factorise into small prob because of the structure of model

$$= \sum_{I=1}^{100} P(x_1 \mid I)\, P(x_2 \mid I) \cdot P(x_5 \mid I)\, P(I)$$

From Huge joint prob table, we have much simpler table to work with.

Summary of Latent Variable

Pros:

i) Introducing latent variables can simplify our model
ii) They can be sometimes meaningful

Cons:

i) Hard to work with, require lots of maths

Probabilistic Clustering

In this lecture, we will use latent variable model for clustering.

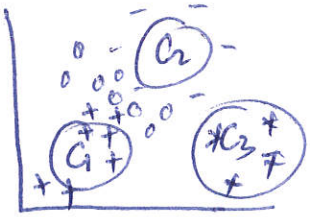
Soft Clustering

* A point can be assigned to multiple clusters using probabilities.

* Hard clustering
  A point is assigned single cluster
  cluster_index = f(x)

* Why do we need to perform soft clustering probabilistically?
  i) Because we want to deal with missing values/features
  ii) We want to tune hyperparameter

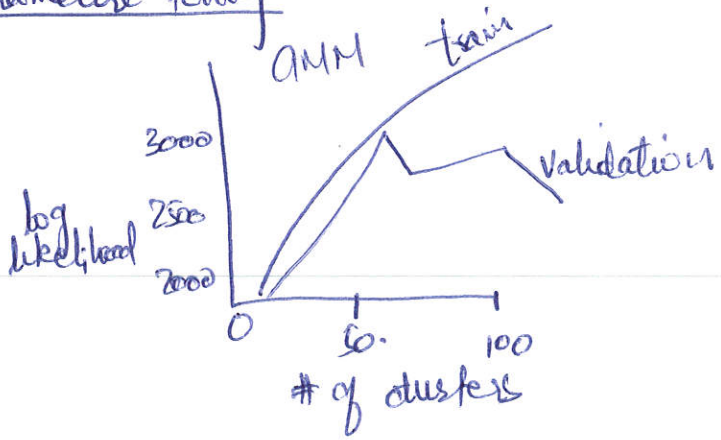
Soft Clustering

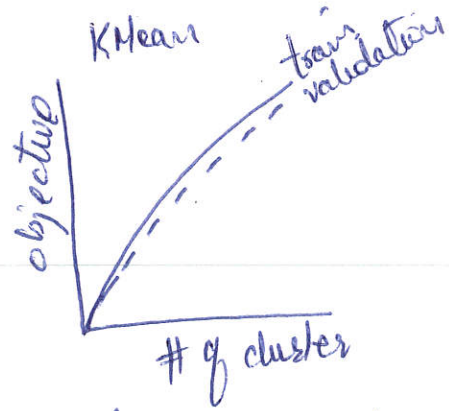


Each point "x" will have $p(C_1|x)$ $p(C_2|x)$ $p(C_3|x)$

i.e $\longrightarrow$ $P(\text{cluster index}|x)$

Hyperparameter tuning





(K-Means will assume the more the cluster, the better is performance)

Another reason we would like to train clustering probabilistically is the we may want to get generative model of our data. It can generate data (like images).

Summary

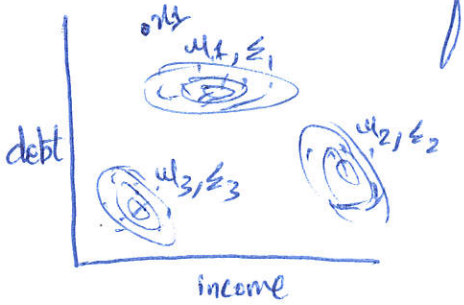
Soft clustering allows us

i) to ~~generate~~ sample data (by getting generative model of data)

ii) to tune hyperparameters

W2-P1-L3 GMM

We will build latent variable model for clustering probabilistically



In the above problem, we will use three Gaussians. Therefore we can assume that each datapoint comes from a weighted sum of each of these three gaussians:

$$p(x|\theta) = \pi_1 \mathcal{N}(x|\mu_1, \Sigma_1) + \pi_2 \mathcal{N}(x|\mu_2, \Sigma_2) + \pi_3 \mathcal{N}(x|\mu_3, \Sigma_3)$$

i.e the density of each data point equals to the sum of three gaussian distributions weighted

$$\pi_1 + \pi_2 + \pi_3 = 1$$

$$\theta = \{\pi_1, \pi_2, \pi_3, \mu_1, \mu_2, \mu_3, \Sigma_1, \Sigma_2, \Sigma_3\}$$

This is called Gaussian Mixture model, where # of mixture = 3

How to find the parameters of GMM?
We can apply maximum likelihood estimation here

$$\max_{\theta} \underbrace{P(X|\theta)}_{\text{Density of dataset } X \text{ given } \theta} = \prod_{i=1}^{N} P(x_i|\theta)$$

$$= \prod_{i=1}^{N} \pi_1 N(x_i|\mu_1,\Sigma_1) + \pi_2 N(x_i|\mu_2,\Sigma_2) + \pi_3 N(x_i|\mu_3,\Sigma_3)$$

s.t i) $\pi_1 + \pi_2 + \pi_3 = 1$ (This will ensure valid Prob. distribution)
ii) $\pi_K > 0$; $K = 1,2,3$
and iii) $\Sigma_K > 0$ $(K=1,2,3)$ PSD-matrix

* Thus GMM is a flexible Probability distribution model.

From last figure

$$P(x_1|\theta) = \pi_1 N(x_1|\mu_1,\Sigma_1) + \pi_2 N(x_1|\mu_2,\Sigma_2) + \pi_3 N(x_1|\mu_3,\Sigma_3)$$

for x_1 →
→ highest
→ potentially x_1 is highest here

→ low
→ low π_2

→ least
→ least π_3

Training GMMs

How do we train GMM using a fact that we would like to avoid using stochastic gradient descent. We will introduce latent-variable that will help us using EM.

From last lecture, density of datapoint "x" was given as follows:

$$\text{Eq A} \longleftarrow P(x|\theta) = \pi_1 \mathcal{N}(x|\mu_1, \Sigma_1) + \pi_2 \mathcal{N}(x|\mu_2, \Sigma_2) + \pi_3 \mathcal{N}(x|\mu_3, \Sigma_3)$$

lets assume there is a latent-variable "t" that causes "x"

$$(t) \longrightarrow (x)$$

In our case, this latent variable will be the cluster index. we will not observe latent variable during testing and training. Given this model, it is safe to assume that prior distribution on "t" is

i.e: $P(t=c|\theta) = \pi_c$ where $c = 1, 2, 3$

And $P(x|t=c, \theta) = \mathcal{N}(x|\mu_c, \Sigma_c)$ If we know from which cluster "x" is coming from

Now, using sum rule (marginalize "t"), we would like to get $P(x|\theta)$ — the likelihood

$$P(x|\theta) = P(x|t=1,\theta) + P(x|t=2,\theta) + P(x|t=3,\theta)$$

($P(t=1|\theta)$, $P(t=2|\theta)$, $P(t=3|\theta)$)

$$\text{Eq B} \longleftarrow P(x|\theta) = \sum_{c=1}^{3} P(x|t=c,\theta)\, P(t=c|\theta)$$

$$P(x|\theta) = P(x, t=1|\theta) + P(x, t=2|\theta) + P(x, t=3|\theta)$$

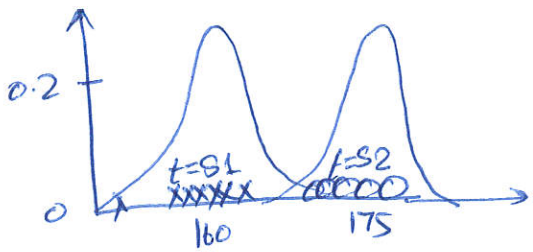
i.e

$$P(x|\theta) = P(x, t=1|\theta) + P(x, t=2|\theta) + P(x, t=3|\theta)$$

$$\Rightarrow P(x|\theta) = \pi_1 \mathcal{N}(x|\mu_1, \Sigma_1) + \pi_2 \mathcal{N}(x|\mu_2, \Sigma_2) + \pi_3 \mathcal{N}(x|\mu_3, \Sigma_3)$$

So, we have seen that Eq A is exactly similar to Eq B therefore we can proceed forward with this latent variable model in Eq B. It will give us the same result as of Eq (A).

Expectation Maximization



How to compute parameters "θ"?

Solution-1 (sources are known) → Hard assignment / → Soft assignment

If we know the sources, i.e we know from which Gaussian data "x" is coming from & can easily evaluate parameters as follows:

$$P(x \mid t=S1, \theta) = N(x \mid \mu_1, \sigma_1^2) \longrightarrow \text{Data given source \& } \theta$$

$$\mu_1 = \frac{\sum_{S_1} x_{S_1}}{\#\text{ of } S_1 \text{ point}} \qquad \sigma_i^2 = \frac{\sum_{S_1} (x_{S_1} - \mu_{S_1})^2}{\#\text{ of } S_1 \text{ points}}$$

This is the example of hard assignment. And if we have soft assignments i.e some posterior distribution on "t" (In this case any point can belong to all clusters but with some probability)

Known for "j": $P(t_i=1 \mid x_i, \theta)$ & $P(t_i=2 \mid x_i, \theta)$ → Posterior on "t"

Then we can estimate the ~~posteriors~~ parameters as follows:

$$\mu_1 = \frac{\sum_i P(t_i=1 \mid x_i, \theta) x_i}{\sum_i P(t_i=1 \mid x_i, \theta)} \qquad \sigma_1^2 = \frac{\sum_i P(t_i=1 \mid x_i, \theta)(x_i - \mu_1)^2}{\sum_i P(t_i=1 \mid x_i, \theta)}$$

Then → Eq(c)

Solution-2

If we know the ~~sources~~ parameters "θ", then we can estimate the ~~Gaussian~~ sources using Bayes Rule.

i.e Given: $P(x \mid t=1, \theta) = N(-2, 1)$ i.e $\mu=2$, $\sigma^2=1$ } for $t=1$

Then using Bayes Rule:

$P(t=1 \mid x, \theta)$ can be imputed

From Bayes Rule, the soft assignment to data "x" can be imputed as follows:

$$\underbrace{P(t=1 \mid x, \theta)}_{\text{posterior probability}} = \frac{P(x, t=1 \mid \theta)}{Z} \leftarrow \text{Joint}$$

$Z \rightarrow$ normalizing constant

$$= \frac{P(x \mid t=1, \theta)\, P(t=1 \mid \theta)}{Z} = \frac{(\text{likelihood}) \times (\text{prior})}{Z}$$

$\gamma = P(t=1,\theta) + P(t=2,\theta)$ → just check

EM Algorithm:

1) Need Gaussian param (μ, Σ) to estimate source $P(t|x,\theta)$
2) Need sources $P(t|x,\theta)$ to estimate gaussian parameters using eq(C).

EM Algo

1) Initial params (μ_1, Σ_1) randomly & (μ_2, Σ_2)

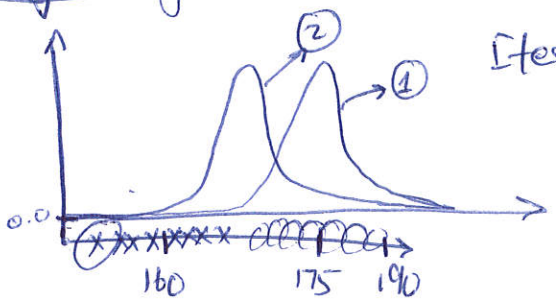
2) Until Convergence:

~~$P(t=c|x_i,\theta)$~~

a) $P(t=1|x_i,\theta) = ??$
$P(t=2|x_i,\theta) = ??$
} for all data points "i"

b) Using $P(t|x_i,\theta)$:
Update "θ" using Eq(C)

Example GMM-training using EM



Iteration - 1

Step-1

i) Initialize "θ" for $K=2$. Now parameters are known so compute:

$$p(t_i=1|x_i,\theta) \;\&\; p(t_i=2|x_i,\theta) \text{ for all points}$$

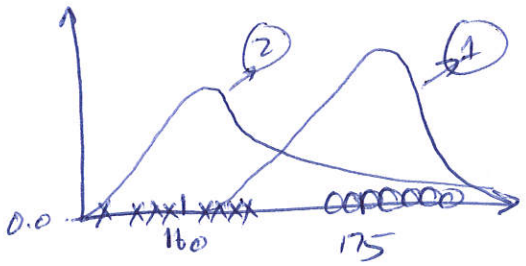
$$p(t_i=1|x_i,\theta) \approx \frac{0.01 \leftarrow \text{density of } (x) \text{ using } \theta_1}{0.01+0.00002 \leftarrow \text{sum of densities}} \approx 1$$

Step-2

i) Once sources are known, re-estimate params ~~using Bayes Rule~~ using formula

$\mu_1 =$ ___ $\Sigma\theta_1 =$ ___

$\mu_2 =$ ___ $\Sigma\theta_2 =$ ___



Iteration - 2

* GMM is flexible probabilistic approach for soft clustering. You can also sample new data points.
* EM can replace Stochastic Gradient.
* EM gives local maxima (the exact solution is NP-hard) for this clustering problem.

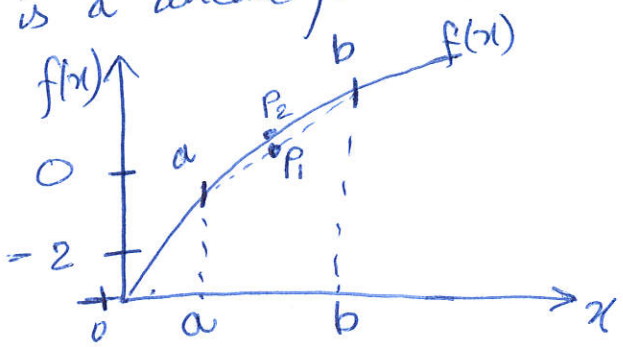
General form of EM

We will derive general form of EM-Algorithm that will help us train almost any latent variable model we can think of. But we will see some mathematical tools first:

1) Concave functions
2) Jensen Inequality

Concave function

If $f(x)$ is a concave function



Then for any a, b & α:

$$f(\alpha a + (1-\alpha) b) \geq \alpha f(a) + (1-\alpha) f(b)$$

and $0 \leq \alpha \leq 1$

i.e point $(P_2) \geq$ point (P_1) whenever $0 \leq \alpha \leq 1$ for two points "a" & "b" on concave function, $f(x)$

and for any number of points on concave functions (like logarithmic function), you can prove the same property:

For e.g: For any three points a_1, a_2, a_3 with weights α_1, α_2 and α_3 we can prove $f(\alpha_1 a_1 + \alpha_2 a_2 + \alpha_3 a_3) \geq \alpha_1 f(a_1) + \alpha_2 f(a_2) + \alpha_3 f(a_3)$

$$\text{s.t } \alpha_1 + \alpha_2 + \alpha_3 = 1$$

We can think of above property in probabilistic term: first we can think of α's as probabilities (α's are weights for each point). Then we can assume latent variable "t" with the following probability distribution

$$P(t=a_1) = \alpha_1$$
$$P(t=a_2) = \alpha_2$$
$$P(t=a_3) = \alpha_3$$

Now we can rewrite Jensen's Inequality as follows:

$$f(\underbrace{\alpha_1 a_1 + \alpha_2 a_2 + \alpha_3 a_3}_{E_{p(t)}t}) \geq \underbrace{\alpha_1 f(a_1) + \alpha_2 f(a_2) + \alpha_3 f(a_3)}_{E_{p(t)}f(t)}$$

i.e $E_{p(t=1)} f(t=1) = \underbrace{\alpha_1 f(a_1)}_{\text{Expected value of } f(t=1)}$

This sum's up Jensen's Inequality:

* For any concave function f, and for any probability distribution "$p(t)$" on "t":

$$f(E_{p(t)}t) \geq E_{p(t)} f(t)$$

"f" of expected value t | Expected value of $f(t)$

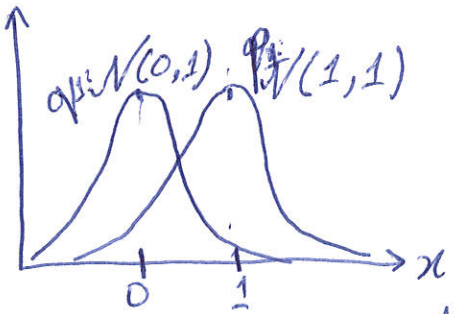
This is true for any number of points "t". For infinite number of points "t", this summation will turn into integral as follows: $f[E_{p(t)}t) \geq \int E_{p(t)} f(t)\, dt$ (check it)

Final thing we need to know is KL-Divergence.

Kullback-Leibler divergence

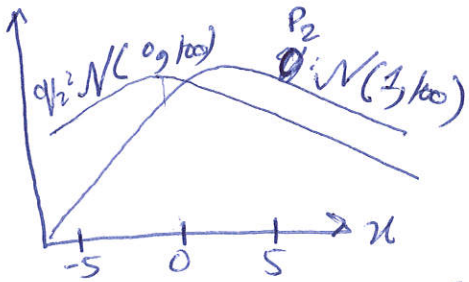
It calculates the difference b/w two probabilistic dist.

Motivation Consider following Gaussians:



i) Difference in parameters = 1

ii) $KL(q_1 \| p_1) = 0.5$



i) Difference in params = 1

ii) $KL(q_2 \| p_2) = 0.005$

* Intuitively, Gaussians on right picture are more similar which is captured by KL-Divergence but not by difference in parameters.

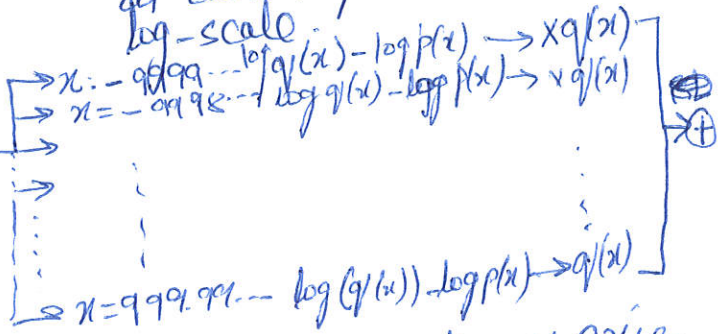
$$KL(q||p) = \int q(x) \log \frac{q(x)}{p(x)} dx = \int \mathbb{E}_q \left[\log \left(\frac{q}{p}\right)\right] dx$$

Expected value of logarithm of ratio.

$\log \frac{q(x)}{p(x)} = \log q(x) - \log p(x)$ → log of ratios measures how different these distributions are at current point "x" in the log-scale

Illustration

$\int q(x) \log \frac{q(x)}{p(x)} dx$ →



i.e we are taking differences of $q(x)$ & $p(x)$ on entire x-axis, and multiplying these differences by $q(x)$

Summary $KL(q||p) = \int q(x) \log \frac{q(x)}{p(x)} dx$

1. $KL(q||p) \neq KL(p||q)$
   i.e its not a proper distance measure in strict mathematical sense

2. $KL(q||q) = 0$
   i.e $KL(q||q) = \int q(x) \log \frac{q(x)}{q(x)} dx$
   $= \int q(x) \log(1) dx$
   $= 0$

3. $KL(q||p) \geq 0$
   To prove this: we should consider

$-KL(q||p)$.

As $KL(q||p) = E_q(\log \frac{q}{p})$

$\Rightarrow -KL(q||p) = E_q(-\log(\frac{q}{p}))$

$= E_q(\log \frac{p}{q})$

Using Jensen's Inequality:

$E_q(\log \frac{p}{q}) \leq \log(E_q \frac{p}{q})$ since log is concave

So, ~~And~~ $\log(E_q \frac{p}{q}) \Rightarrow$

$$E_q\left(\frac{p}{q}\right) = \int q(x) \frac{p(x)}{q(x)} dx$$

$$\Rightarrow \log E_q\left(\frac{p}{q}\right) = \log \int q(x) \frac{p(x)}{q(x)} dx$$

$$= \log \int p(x) dx \quad (\int p(x) dx = 1)$$

$$= \log(1)$$

$$= 0$$

$$\Rightarrow \boxed{\log E_q\left(\frac{p}{q}\right) = 0} \rightarrow \text{eq} (A)$$

This means: ~~$KL(q||p) \geq 0$~~

$$-KL(q||p) = E_q\left(-\log \frac{q}{p}\right)$$

$$= E_q\left(\log \frac{p}{q}\right)$$

Using Jensen's Inequality:

$$E_q\left(\log \frac{p}{q}\right) \leq \log\left(E_q \frac{p}{q}\right)$$

And using (A)

$$E_q\left[\log \frac{p}{q}\right] \leq 0$$

i.e $-KL(q||p)$ is alway non-positive

which implies $KL(q||p) \geq 0$ □

GENERAL form of Expectation Maximization

Motivation Lets assume we have latent variable "t" for each "x":

So, the marginal of "x_i" is given as:

$$p(x_i|\theta) = \sum p(x_i, t_i|\theta) \rightarrow \text{Joint over } x_i, t_i$$

If $t_i \in [1, 2, 3]$

$$= \sum_{c=1}^{3} p(x_i, t_i = c|\theta) \rightarrow \text{Marginalize "}t_i\text{" using sum-rule}$$

$$= \sum_{c=1}^{3} p(x_i|t_i = c, \theta)\, p(t_i = c|\theta)$$

likelihood, prior

Our problem here is:

$$\max_{\theta} p(X|\theta) \longrightarrow$$ This is marginal likelihood as we have marginalized latent variable "t".

Now: $$\max_{\theta} p(X|\theta) = \max_{\theta} \log p(X|\theta) \max_{\theta} \prod_{i=1}^{N} p(x_i|\theta) \text{ (product)}$$

$$= \max_{\theta} \log \left(\prod_{i=1}^{N} p(x_i|\theta) \right)$$

$$= \max_{\theta} \log p(X|\theta)$$

And $$\max_{\theta} \log p(X|\theta) = \log \prod_{i=1}^{N} p(x_i|\theta)$$ (sums) } to optimize

$$= \sum_{i=1}^{N} \log p(x_i|\theta)$$

Now: $$\log p(X|\theta) = \sum_{i=1}^{N} \log p(x_i|\theta)$$

Adding latent variables here

$$\log p(X|\theta) = \sum_{i=1}^{N} \log \sum_{c=1}^{3} p(x_i, t_i = c|\theta)$$

Now, $\max_{\theta} p(X|\theta)$ likelihood function is transformed into the following with latent variables "t".

$$\log p(X|\theta) = \sum_{i=1}^{N} \log \sum_{c=1}^{3} p(x_i, t_i = c|\theta)$$

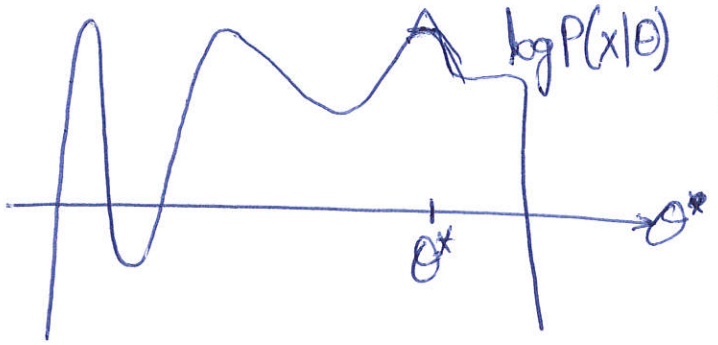
This is what we would like to maximize. We can use stochastic gradient descent here.

Now lets derive the lower bound of this function using Jensen's Inequality. Lets assume we came up with lower bound $L(\theta)$, then using Jensen's Inequality we know.

$$\sum_{i=1}^{N} \log \sum_{c=1}^{3} p(x_i, t_i = c|\theta) \geq L(\theta)$$ → Here $L(\theta)$ is easy to optimize

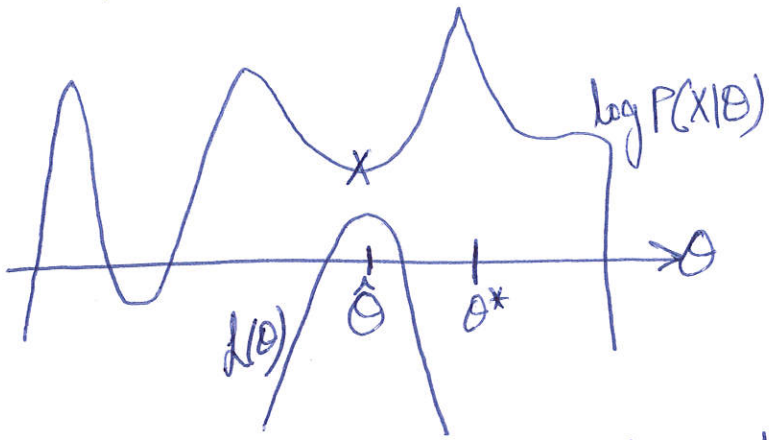
So, instead of maximizing $\sum_{i=1}^{N} \log \sum_{c=1}^{3} p(x_i, t_i = c|\theta)$ we can maximize $L(\theta)$

Lets assume that our marginal loglikelihood look as follows:
$\log P(X|\theta)$



(*) θ^* is the optimum of $\log P(X|\theta)$

Lets calculate $L(\theta)$ for $\log P(X|\theta)$. Its value will be $\leq \log P(X|\theta)$ at all ~~points~~ "θ". Let assume we come up with the following $L(\theta)$



The optimum of $L(\theta)$ is at $\hat{\theta}$ but we know that at $\theta = \hat{\theta}$, we get local minimum of $\log P(X|\theta)$ — our job was to find local maximum of $\log P(X|\theta)$. So, now lets generate family of lower bounds as follows: Lets introduce any distribution "q" on our latent variable "t" as: $q(t_i = c)$

Now multiply and divide $\log P(X|\theta)$ by $q(t_i = c)$

$$\Rightarrow \log p(X|\theta) = \sum_{i=1}^{N} \log \sum_{c=1}^{3} \frac{q(t_i=c)}{q(t_i=c)} p(x_i, t_i = c|\theta)$$

$$= \sum_{i=1}^{N} \log \sum_{c=1}^{3} q(t_i=c) \underbrace{\frac{p(x_i, t_i=c|\theta)}{q(t_i=c)}}_{\alpha_3} \quad \rightarrow v_3$$

$$= \sum_{i=1}^{N} \log \left[\underbrace{q(t_i=1)}_{\alpha_1} \underbrace{\frac{p(x_i, t_i=1|\theta)}{q(t_i=1)}}_{v_1} + \cdots + \frac{q(t_i=3)\, p(x_i, t_i=3|\theta)}{q(t_i=3)} \right]$$

And we know that $\sum q(t_i) = 1$

i.e $\alpha_1 + \alpha_2 + \alpha_3 = 1$

Thus

$$\log P(X|\theta) = \sum_{i=1}^{N} \log \left[\alpha_1^i V_1^i + \alpha_2^i V_2^i + \alpha_3^i V_3^i \right]$$

where $\alpha_1^i = q(t_i = 1)$

$$V_1^i = \frac{P(x_i, t_i = 1|\theta)}{q(t_i = 1)}$$

from Jensen's Inequality

$$\log\left(\sum_c \alpha_c V_c \right) \geq \sum_c \alpha_c \log(V_c)$$

Therefore, $\sum_c \alpha_c \log(V_c)$ is the lower bound of $\log(\sum_c \alpha_c V_c)$.
This means, our lower bound of $\log P(X|\theta)$ is given as follows:

$$\log P(X|\theta) = \sum_{i=1}^{N} \log \left[\sum_{c=1}^{3} \underbrace{q(t_i = c)}_{\alpha_c} \underbrace{\frac{P(x_i, t_i = c|\theta)}{q(t_i = c)}}_{V_c} \right]$$

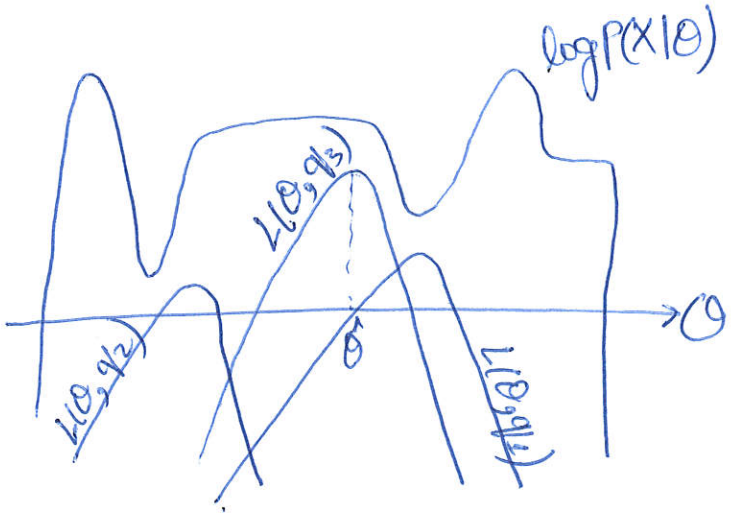
$$\geq \boxed{\sum_{i=1}^{N} \sum_{c=1}^{3} \underbrace{q(t_i = c)}_{\alpha_c} \log \underbrace{\frac{P(x_i, t_i = c|\theta)}{q(t_i = c)}}_{V_c}} \quad \Phi$$

$\hookrightarrow L(\theta, q)$

Now, using different values of "q", we can obtain family of lower bounds for $\log P(X|\theta)$ given parameter "θ".

Now our picture looks like this:

$$\log P(X|\theta) \geq L(\theta, q) \quad \text{for any } q$$



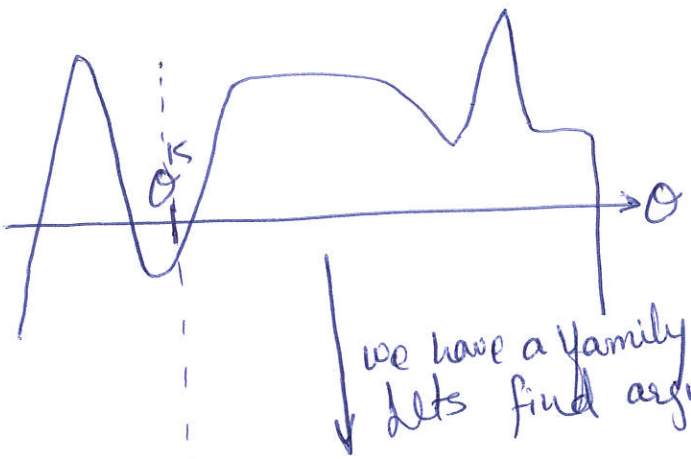
Here, we can choose $L(\theta, q_3)$, get its maximum at $\hat{\theta}$ and obtain $\log P(X|\theta=\hat{\theta})$ which will be a good approximation of optimum for $\log P(X|\theta)$.

E-STEP: $q^{K+1} = \underset{q}{\operatorname{argmax}}\, L(\theta^K, q)$ at K^{th} iteration

M-STEP: $\theta^{K+1} = \underset{\theta}{\operatorname{argmax}}\, L(\theta, q^{K+1})$

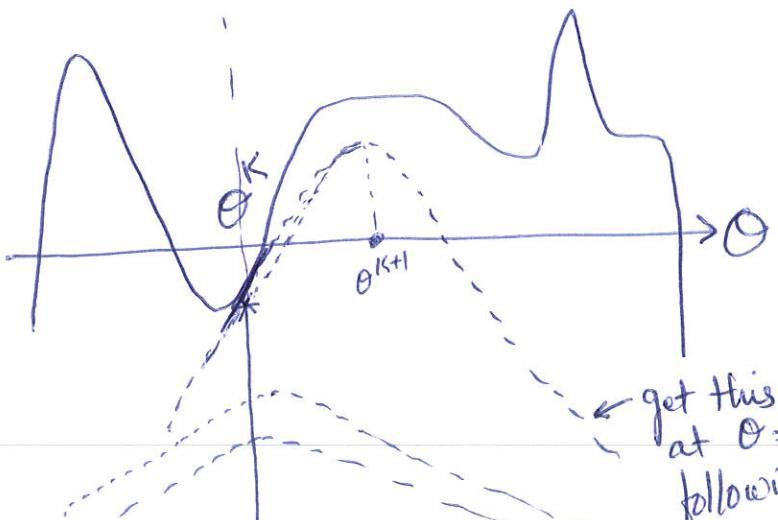
VISUALIZATION

Iteration - 0: Let initialize θ^K



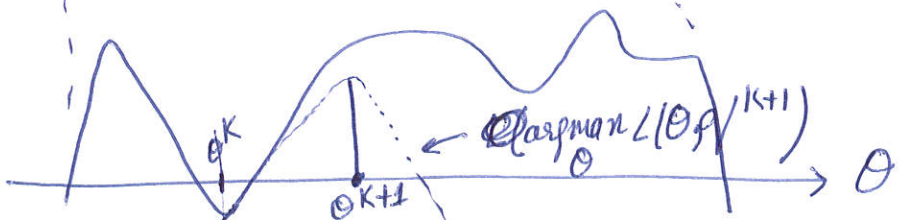
we have a family of loose bound $L(\theta, q)$.
Lets find $\underset{q}{\operatorname{argmax}}\, L(\theta^K, q)$

E-STEP



get this loose bound because at $\theta = \theta^K$ it fullfills the following requirement
$q^{K+1} = \underset{q}{\operatorname{argmax}}\, L(\theta^K, q)$

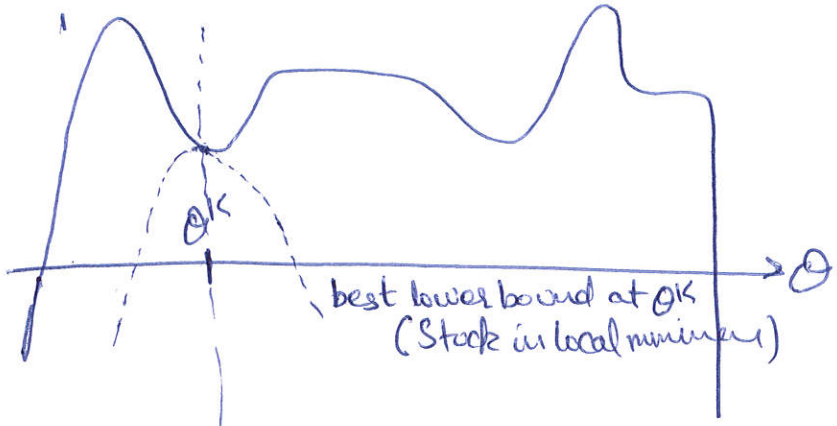
M-STEP:



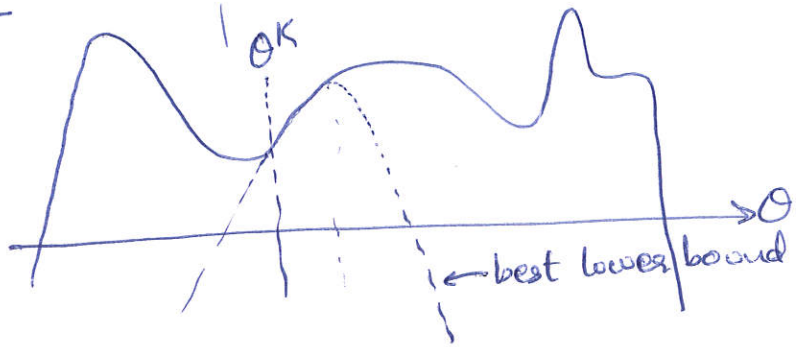
$\ln \log p(X|\theta) \geq \mathcal{L}(\theta, q)$ for any q

* $\mathcal{L}(\theta, q)$ is called variational lower bound because you can vary it by varying "q" & "q" was not the part of original model.

(1)

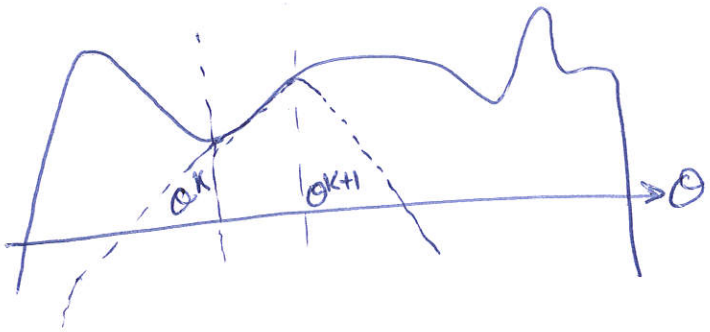


Restart

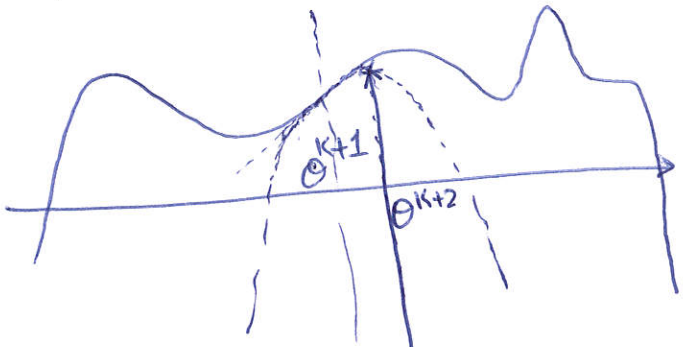


$q^{K+1} = \arg\max_q L(\theta^K, q)$

Get the $\max_\theta L(\theta, q^{K+1})$



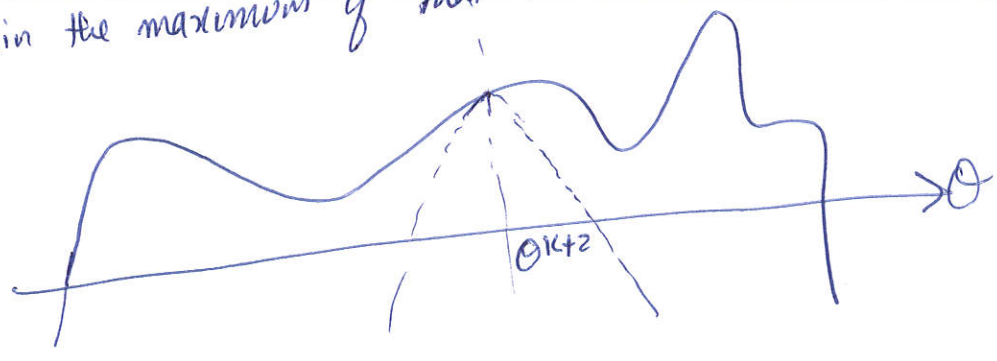
$\theta^{K+1} = \arg\max_\theta L(\theta, q^{K+1})$



$q^{K+2} = \arg\max_q L(\theta^{K+1}, q)$

(out of all lower bounds, this gives highest value of $L(\theta^{K+1})$)

Now, obtain the maximum q that lower bound in θ.



E-STEP:

$$q^{k+1} = \underset{q}{\operatorname{argmax}}\, \mathcal{L}(\theta^k, q)$$

Find the best lower bound

M-STEP:

$$\theta^{k+1} = \underset{\theta}{\operatorname{argmax}}\, \mathcal{L}(\theta, q^{k+1})$$

Get the max of that lower bound

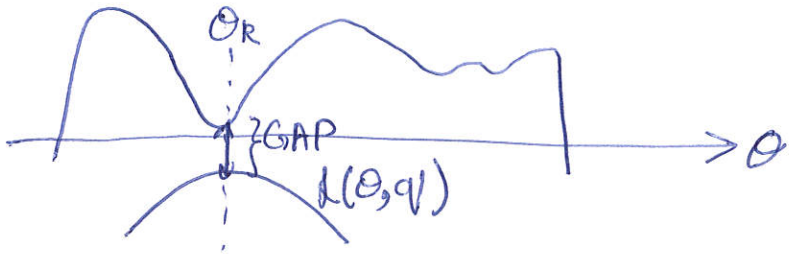
E-STEP DETAILS

We Now know that:

$$\log P(X|\theta) \geq \underbrace{\mathcal{L}(\theta, q)}_{\text{variational "lower bound which depend on } q\text{". "q" is itself a distribution.}}$$

In E-STEP:

$$q^{K+1} = \underset{q}{\operatorname{argmax}}\ \mathcal{L}(\theta^K, q) \quad \left\{ \text{find the best lower bound} \right.$$



$$\text{GAP} = \underbrace{\log P(X|\theta)}_{\text{Marginal log likelihood}} - \underbrace{\mathcal{L}(\theta, q)}_{\text{variational lower bound}}$$

} E-STEP tries to minimize this gap.

$$= \sum_{i=1}^{N} \log P(x_i|\theta) - \sum_{i=1}^{N} \sum_{c_i=1}^{3} q(t_i=c) \log \frac{P(x_i, t_i=c|\theta)}{q(t_i=c)}$$

$$= \sum_{i=1}^{N} \left[\underbrace{\log P(x_i|\theta)}_{\text{Independent of } c} \times \underbrace{\sum_{c_i=1}^{3} q(t_i=c)}_{=1} - \sum_{i=1}^{N} \sum_{c_i=1}^{3} q(t_i=c) \frac{\log P(x_i, t_i=c|\theta)}{q(t_i=c)} \right.$$

$$= \sum_{i=1}^{N} \left[\sum_{c_i=1}^{3} \log P(x_i|\theta)\, q(t_i=c) - \sum_{i=1}^{N} \sum_{c_i=1}^{3} q(t_i=c) \frac{\log P(x_i, t_i=c|\theta)}{q(t_i=c)} \right]$$

$$= \sum_{i=1}^{N} \sum_{c_i=1}^{3} q(t_i=c) \left[\log P(x_i|\theta) - \log \frac{P(x_i, t_i=c|\theta)}{q(t_i=c)} \right]$$

$$= \sum_{i=1}^{N} \sum_{c_i=1}^{3} q(t_i=c) \left[\log \frac{P(x_i|\theta)}{P(x_i, t_i=c|\theta)} \times q(t_i=c) \right]$$

IMP Simplification (Quiz)

$$\frac{P(x_i|\theta)}{P(x_i, t_i|\theta)} = \frac{P(x_i|\theta)}{P(t_i|x_i, \theta)\, p(x_i|\theta)} = \frac{1}{P(t_i|x_i, \theta)}$$

using definition of conditional probability.

$$p(x_i, t_i = c|\theta) = \underbrace{p(t_i = c|x_i, \theta)}_{\text{conditional}} \underbrace{p(x_i|\theta)}_{\text{prior distribution}}$$

$$= \sum_{i=1}^{N} \sum_{c=1}^{3} q(t_i = c) \log \frac{p(x_i|\theta) q(t_i = c)}{p(t_i = c|x_i, \theta) p(x_i|\theta)}$$

$$= \sum_{i=1}^{N} \sum_{c=1}^{3} q(t_i = c) \log \frac{p(t_i = c)}{p(t_i = c|x_i, \theta)}$$

$$= KL\left(q(t_i) \,\|\, p(t_i = c|x_i, \theta)\right)$$

$$\Rightarrow GAP = \log p(X|\theta) - \mathcal{L}(\theta, q)$$

$$= \sum_{i=1}^{N} KL\left(q(t_i) \,\|\, \underbrace{p(t_i|x_i, \theta)}_{\text{posterior distribution}}\right)$$

→ We want to minimize GAP
- → we want to maximize $\mathcal{L}(\theta, q)$ w.r.t q
- → this means we have to $\min_q \left[-\mathcal{L}(\theta, q)\right]$

$$GAP = \underbrace{\log p(X|\theta)}_{\text{Independent of } q} - \underbrace{\mathcal{L}(\theta, q)}_{\substack{\max_q \\ \min_q}}$$

Thus $\max_q \mathcal{L}(\theta, q)$ is equivalent to minimizing $\log p(X|\theta) - \mathcal{L}(\theta, q)$.
Thus minimizing $\log p(X|\theta) - \mathcal{L}(\theta, q)$ is equivalent to minimizing $\sum_{i=1}^{N} KL\left(q(t_i) \,\|\, p(t_i|x_i, \theta)\right)$ or sum of KL divergences.
This implies $\max_q \mathcal{L}(\theta, q)$ is equivalent to $\min_q \sum_{i=1}^{N} KL\left[q(t_i) \,\|\, p(t_i|x_i, \theta)\right]$

Recall that KL-divergence had two properties:
i) $KL(q\|p) \geq 0$
ii) $KL(q\|q) = 0$

Thus $\sum_{i=1}^{N} KL\left[q(t_i) \,\|\, p(t_i|x_i, \theta)\right] \geq 0$ (& it can never be negative)

In order to $\min_q \sum_{i=1}^{N} KL\left[q(t_i) \,||\, p(t_i|x_i,\theta)\right]$,

we will set $q(t_i) = \underbrace{p(t_i|x_i,\theta)}_{\text{posterior distribution}}$ as $KL(q||q) = 0$. It will return us global minimum in "q"

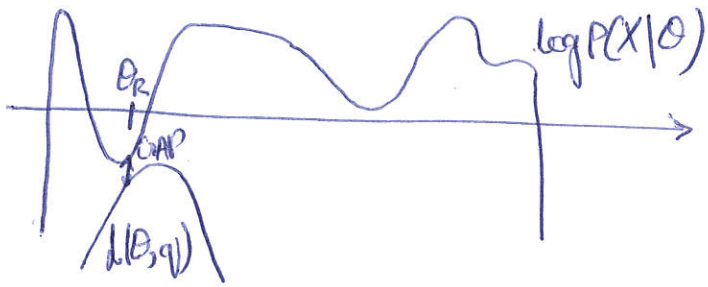
Outcome

Minimizing sum of KL-divergence for E-step simply asks us to set $q(t_i) = p(t_i|x_i,\theta)$. It will return us the best lower-bound $L(\theta^K, q)$ at $\theta = \theta^K$

E-Step

We know that: $\log p(X|\theta) \geq L(\theta, q)$

E-step: $q^{K+1} = \underset{q}{\operatorname{argmax}}\, L(\theta^K, q)$



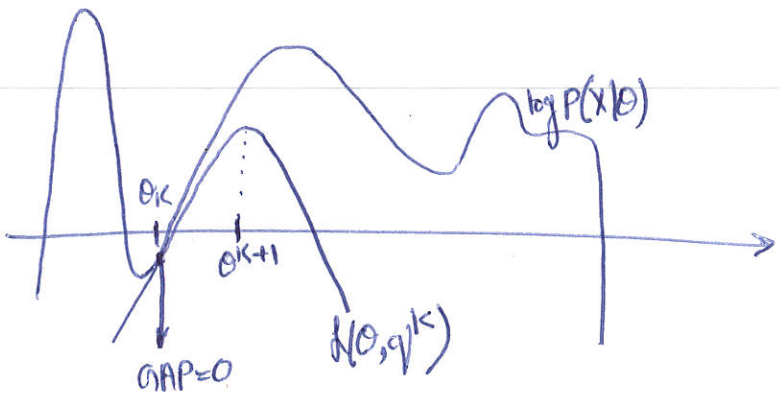
Conclusion To get the best lower bound, we define gap:

$$\text{gap: } \log p(X|\theta) - L(\theta, q) = \sum_i KL\left[q(t_i) \,||\, p(t_i|x_i,\theta)\right]$$

To set gap = 0, use $KL(q||q) = 0 \Rightarrow$ set $q(t_i) = p(t_i|x_i,\theta)$

This will minimize gap but maximize lower bound $L(\theta^K, q)$.

E-STEP: $\underset{q(t_i)}{\operatorname{argmax}}\, L(\theta^K, q) = p(t_i|x_i,\theta)$



W2-P2-24

M-STEP

In M-Step, we want to maximize variational lower bound $\mathcal{L}(\theta, q)$ w.r.t "θ"

$$\mathcal{L}(\theta, q) = \sum_i \sum_c q(t_i=c) \log \frac{P(x_i, t_i=c|\theta)}{q(t_i=c)}$$

$$= \underbrace{\sum_i \sum_c q(t_i=c) \log P(x_i, t_i=c|\theta)}_{E_q \log P(X,T|\theta)} - \underbrace{\sum_i \sum_c q(t_i=c) \log q(t_i=c)}_{\text{constant w.r.t "}\theta\text{"}}$$

$E_q \log P(X,T|\theta)$ = expected value of log of joint distribution of X & T w.r.t variational distribution "q".

$$= \underbrace{E_q \log P(X,T|\theta)}_{} + \text{const}$$

we choose $P(X,T|\theta)$ such that its log becomes concave and $E_q[\text{concave function}]$ is also concave. So you can easily optimize it. It ensures that this function has global maxima

SKETCH of EM-Algorithm

for $K = 1 \ldots$ until convergence

E-STEP: $q^{K+1} = \underset{q}{\text{argmax}}\ \mathcal{L}(\theta^K, q)$

$$= \underset{q}{\text{argmin}}\ KL\left[\underbrace{q(T)}_{\text{variational distribution}} \| \underbrace{P(T|X,\theta^K)}_{\text{posterior distribution}}\right]$$

$$\Leftrightarrow \quad q^{K+1}(t_i) = P(t_i|x_i, \theta^K)$$

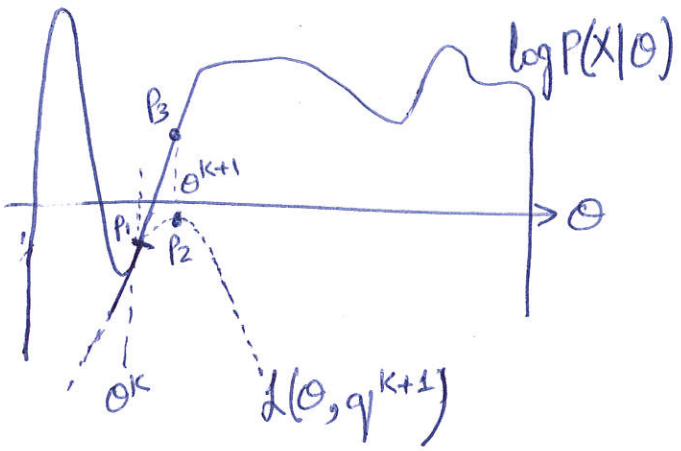
M-STEP:

$$\theta^{K+1} = \underset{\theta}{\text{argmax}}\ \mathcal{L}(\theta, q^{K+1})$$

$$= \underset{\theta}{\text{argmax}}\ \cancel{E_q \log P(x_i, t_i|\theta)}$$

$$= \underset{\theta}{\text{argmax}}\ \underbrace{E_q^{K+1} \log P(X,T|\theta)}_{\text{Expected value of joint distribution}}$$

Convergence of EM



$$\underbrace{\log P(X|\theta^{K+1})}_{P_3} \geq \underbrace{L(\theta^{K+1}, q^{K+1})}_{P_2} \geq \underbrace{L(\theta^{K}, q^{K+1})}_{P_1}$$

Can we compare value of $L(\theta^K, q^{K+1})$ with log-likelihood $\log P(X|\theta^K)$ in general case?

Ans: Yes, since for any "q" the following inequality holds true $L(\theta^K, q) \leq \log P(X|\theta^K)$, but for q^{K+1} is the maximizer of the lower bound $L(\theta^K, q^{K+1}) = \max L(\theta^K, q)$ and in the best case it became as large as $\log P(X|\theta^K)$ ∀

Hence $\underbrace{\log P(X|\theta^{K+1})}_{P_3} \geq \cancel{L(\theta^K, q^{K+1})} \underbrace{\log P(X|\theta^K)}_{P_1}$

i.e the marginal log likelihood never decreases during an iteration of EM.

- At each iteration, EM doesn't decrease objective (good for debugging)
- Guaranteed to converge to a local maximum (or saddle point).

W2-P2-L5

(4)

Example: EM-for discrete mixture, E-STEP: