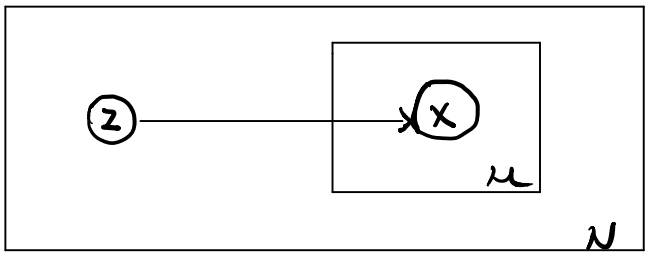
EECS 445 HW4 Juncheng An

1. 1.0 (a)



(b) learning parameters are $\theta_A, \theta_B, \theta_C$

(c) latent variables are all $Z^{(i)}$ which indicate city assignment

1.1 (a) $\log p(\vec{x}^{(i)}, Z^{(i)}; \vec{\theta}) = \log \frac{1}{3} \cdot (\theta_{Z^{(i)}})^{S_i} \cdot (1-\theta_{Z^{(i)}})^{M-S_i}$

$= \log \frac{1}{3} + S_i \cdot \log \theta_{Z^{(i)}} + (M-S_i) \log (1-\theta_{Z^{(i)}})$

- Because for each city, we have $p = \frac{1}{3}$ to choose it and rain or not problem would be a Binomial $(M, \theta_{Z^{(i)}})$ for each city.

(b) · Just multiply all of the trails together

$\log P(X, Z; \vec{\theta}) = \log \prod_{i=1}^{N} p(\vec{x}^{(i)}, \vec{z}^{(i)}; \vec{\theta})$

$= \sum_{i=1}^{N} \log p(\vec{x}^{(i)}, \vec{z}^{(i)}; \vec{\theta})$

$= N \log \frac{1}{3} + S_i \cdot \log \theta_{Z^{(i)}} + (M-S_i) \log (1-\theta_{Z^{(i)}})$

(c) · Denote the log likelihood function as l

$$l = \log p(\lambda, \vec{z}; \vec{\theta})$$

$\Rightarrow$ define ☆

$$I_{Ai} = \begin{cases} 1 & \text{if } z^{(i)} = A \\ 0 & \text{if } z^{(i)} \neq A \end{cases} \quad \forall i$$

$$I_{Bi} = \begin{cases} 1 & \text{if } z^{(i)} = B \\ 0 & \text{if } z^{(i)} \neq B \end{cases} \quad \forall i$$

$$I_{Ci} = \begin{cases} 1 & \text{if } z^{(i)} = C \\ 0 & \text{if } z^{(i)} \neq C \end{cases} \quad \forall i$$

$\therefore I_{Ai}, I_{Bi}, I_{Ci}$ are indicator functions for A, B, C

· $\frac{d}{d\theta_A} l = \sum_{i=1}^{N} [s_i \cdot (\theta_A)^{-1} - (1 - s_i) \cdot (1 - \theta_A)^{-1}] \cdot I_{Ai} = 0$

$$\sum_{i=1}^{N} [s_i (1 - \theta_A) - (1 - s_i) \theta_A] \cdot I_{Ai} = 0$$

$$\sum_{i=1}^{N} [s_i - s_i \cdot \theta_A - \theta_A + s_i \cdot \theta_A] \cdot I_{Ai} = 0$$

$$\hat{\theta}_A = \frac{\sum_{i=1}^{N} s_i \cdot I_{Ai}}{\sum_{i=1}^{N} 1 \cdot I_{Ai}}$$

$\Rightarrow$ exactly same for B and C

$$\hat{\theta}_B = \frac{\sum_{i=1}^{N} s_i \cdot I_{Bi}}{\sum_{i=1}^{N} 1 \cdot I_{Bi}} \quad ; \quad \hat{\theta}_C = \frac{\sum_{i=1}^{N} s_i \cdot I_{Ci}}{\sum_{i=1}^{N} 1 \cdot I_{Ci}}$$

· Intuitively, these 3 mle's $\hat{\theta}_A, \hat{\theta}_B, \hat{\theta}_C$ are just use the total number of sunny days of A, B, C in the sample data devided by the total number of observed sample total days of A, B, C to get a fraction. $\hat{\theta}_i$ is $\frac{\text{total sunny days of } i}{\text{total days of } i}$

· $\hat{\theta}_A = \frac{8+7}{10 \times 2} = 0.75$; $\hat{\theta}_B = \frac{2+3}{2 \times 10} = 0.25$

$\hat{\theta}_C = \frac{11}{2 \times 10} = 0.55$

1.2 (a) $P(Z^{(i)}=k \mid \vec{x}^{(i)}; \vec{\theta}^{(t)}) = \dfrac{P(Z^{(i)}=k, \vec{x}^{(i)}; \vec{\theta}^{(t)})}{P(\vec{x}^{(i)}; \vec{\theta}^{(t)})}$

$$= \frac{P(\vec{x}^{(i)} \mid Z^{(i)}=k; \vec{\theta}^{(t)}) \cdot P(Z^{(i)}=k)}{\sum_{j=A}^{C} P(\vec{x}^{(i)} \mid Z^{(i)}=j; \vec{\theta}^{(t)}) \cdot P(Z^{(i)}=j)}$$

$$\Rightarrow P(Z^{(i)}=k) = P(Z^{(i)}=j) = \frac{1}{3} \quad \forall k, j$$

$$\therefore = \frac{P(\vec{x}^{(i)} \mid Z^{(i)}=k; \vec{\theta}^{(t)})}{\sum_{j=A}^{C} P(\vec{x}^{(i)} \mid Z^{(i)}=j; \vec{\theta}^{(t)})}$$

$$\Rightarrow P(\vec{x}^{(i)} \mid Z^{(i)}=A; \vec{\theta}^{(t)}) = \binom{n}{s_i} \cdot (\theta_A^{(t)})^{s_i} (1-\theta_A^{(t)})^{n-s_i}$$

$$\Rightarrow P(\vec{x}^{(i)} \mid Z^{(i)}=B; \vec{\theta}^{(t)}) = \binom{n}{s_i} \cdot (\theta_B^{(t)})^{s_i} (1-\theta_B^{(t)})^{n-s_i}$$

$$\Rightarrow P(\vec{x}^{(i)} \mid Z^{(i)}=C; \vec{\theta}^{(t)}) = \binom{n}{s_i} \cdot (\theta_C^{(t)})^{s_i} (1-\theta_C^{(t)})^{n-s_i}$$

$$\therefore P(Z^{(i)}=k \mid \vec{x}^{(i)}; \vec{\theta}) = \frac{(\theta_K^{(t)})^{s_i} (1-\theta_K^{(t)})^{n-s_i}}{(\theta_A^{(t)})^{s_i}(1-\theta_A^{(t)})^{n-s_i} + (\theta_B^{(t)})^{s_i}(1-\theta_B^{(t)})^{n-s_i} + (\theta_C^{(t)})^{s_i}(1-\theta_C^{(t)})^{n-s_i}}$$

(b) · Write a python fn to calculate, see appendix.

· $\vec{\theta}^{(0)} = [0.1, 0.4, 0.6]$

① $Z_1^{(1)}$: $P(Z^{(1)}=A \mid x_1; \vec{\theta}) = 0.6316$
$P(Z^{(1)}=B \mid x_1; \vec{\theta}) = 0.02908$ $\therefore \hat{Z}_1^{(1)} = A$
$P(Z^{(1)}=B \mid x_1; \vec{\theta}) = 0.3313$

$Z_1^{(2)}$: $P(Z^{(1)}=A \mid x_1; \vec{\theta}) = 0.03378$
$P(Z^{(2)}=B \mid x_1; \vec{\theta}) = 0.8068$ $\therefore \hat{Z}_1^{(2)} = B$
$P(Z^{(2)}=C \mid x_1; \vec{\theta}) = 0.1593$

$Z_1^{(3)}$: $P(Z^{(3)}=A \mid x_1; \vec{\theta}) = 0.2041$
$P(Z^{(3)}=B \mid x_1; \vec{\theta}) = 0.39745$ $\therefore$ B & C tie $\Rightarrow$ B
$P(Z^{(3)}=C \mid x_1; \vec{\theta}) = 0.39745$ $\therefore \hat{Z}_1^{(3)} = B$

$Z_1^{(4)}$: $P(Z^{(4)}=A \mid \bar{x}; \bar{\theta}) = 0.0109$

$P(Z^{(4)}=B \mid \bar{x}; \bar{\theta}) = 0.0903$ $\therefore \hat{Z}_1^{(4)} = B$

$P(Z^{(4)}=C \mid \bar{x}; \bar{\theta}) = 0.0798$

$Z_1^{(5)}$ $P(Z^{(5)}=A \mid \bar{x}; \bar{\theta}) = 0.3558$

$P(Z^{(5)}=B \mid \bar{x}; \bar{\theta}) = 0.1982$ $\therefore \hat{Z}_1^{(5)} = C$

$P(Z^{(5)}=C \mid \bar{x}; \bar{\theta}) = 0.4459$

$Z_1^{(6)}$ $P(Z^{(6)}=A \mid \bar{x}; \bar{\theta}) = 0.5089$

$P(Z^{(6)}=B \mid \bar{x}; \bar{\theta}) = 0.081$ $\therefore \hat{Z}_1^{(6)} = A$

$P(Z^{(6)}=C \mid \bar{x}; \bar{\theta}) = 0.4101$

$\hat{\theta}_{A1} = \frac{8+7}{20} = 0.75$; $\hat{\theta}_{B1} = \frac{3+5+2}{30} = \frac{1}{3}$

$\hat{\theta}_{C1} = \frac{6}{10} = 0.6$

$\therefore \bar{\theta}^{(1)} = [0.75, 1/3, 0.6]$

$\bar{Z}_1 = [A, B, B, B, C, A]$

② $Z_2^{(1)}$: $P(Z^{(1)}=A \mid \bar{x}; \bar{\theta}) = 0.6942$

$P(Z^{(1)}=B \mid \bar{x}; \bar{\theta}) = 0.0075$ $\therefore \hat{Z}_2^{(1)} = A$

$P(Z^{(1)}=B \mid \bar{x}; \bar{\theta}) = 0.2981$

$Z_2^{(2)}$: $P(Z^{(2)}=A \mid \bar{x}; \bar{\theta}) = 0.0101$

$P(Z^{(2)}=B \mid \bar{x}; \bar{\theta}) = 0.8509$ $\therefore \hat{Z}_2^{(2)} = B$

$P(Z^{(2)}=C \mid \bar{x}; \bar{\theta}) = 0.1389$

$Z_2^{(3)}$: $P(Z^{(3)}=A \mid \bar{x}; \bar{\theta}) = 0.1476$

$P(Z^{(3)}=B \mid \bar{x}; \bar{\theta}) = 0.3452$ $\therefore \hat{Z}_2^{(3)} = C$

$P(Z^{(3)}=C \mid \bar{x}; \bar{\theta}) = 0.5072$

$Z_2^{(4)}$: $P(Z^{(4)}=A \mid \bar{x}; \bar{\theta}) = 0.001$

$P(Z^{(4)}=B \mid \bar{x}; \bar{\theta}) = 0.9466$ $\therefore \hat{Z}_2^{(4)} = B$

$P(Z^{(4)}=C \mid \bar{x}; \bar{\theta}) = 0.0515$

$Z_2^{(5)}$ $P(Z^{(5)}=A \mid \bar{x}; \bar{\theta}) = 0.3218$

$P(Z^{(5)}=B \mid \bar{x}; \bar{\theta}) = 0.1254$ $\therefore \hat{Z}_2^{(5)} = C$

$P(Z^{(5)}=C \mid \bar{x}; \bar{\theta}) = 0.5528$

$Z_2^{(6)}$ $P(Z^{(6)}=A \mid \bar{x}; \bar{\theta}) = 0.519$

$P(Z^{(6)}=B \mid \bar{x}; \bar{\theta}) = 0.0337$ $\therefore \hat{Z}_2^{(6)} = A$

$P(Z^{(6)}=C \mid \bar{x}; \bar{\theta}) = 0.4464$

$\therefore \hat{\theta}_{A2} = \frac{8+7}{20} = 0.75$; $\hat{\theta}_{B2} = \frac{3+2}{20} = 0.25$

$\hat{\theta}_{C2} = \frac{11}{20} = 0.55$

$\therefore \bar{\theta}^{(2)} = [0.75, 0.25, 0.55]$

$\bar{Z}_2 = [A\ B\ C\ B\ C\ A]$

$Z_3^{(1)}$: $P(Z^{(1)}=A \mid \bar{x}; \bar{\theta}) = 0.785$

$P(Z^{(1)}=B \mid \bar{x}; \bar{\theta}) = 0.001$ $\therefore \hat{Z}_3^{(1)} = A$

$P(Z^{(1)}=B \mid \bar{x}; \bar{\theta}) = 0.212$

$Z_3^{(2)}$: $P(Z^{(2)}=A \mid \bar{x}; \bar{\theta}) = 0.00$

$P(Z^{(2)}=B \mid \bar{x}; \bar{\theta}) = 0.763113$ $\therefore \hat{Z}_3^{(2)} = B$

$P(Z^{(2)}=C \mid \bar{x}; \bar{\theta}) = 0.2274$

$Z_3^{(3)}$: $P(Z^{(3)}=A \mid \bar{x}; \bar{\theta}) = 0.16645$

$P(Z^{(3)}=B \mid \bar{x}; \bar{\theta}) = 0.16645$ $\therefore \hat{Z}_3^{(3)} = C$

$P(Z^{(3)}=C \mid \bar{x}; \bar{\theta}) = 0.66$

$Z_3^{(4)}$: $P(Z^{(4)}=A \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.0012b$

$P(Z^{(4)}=B \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.9123$ $\therefore \hat{Z}_3^{(4)} = B$

$P(Z^{(4)}=C \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.0850$

$Z_3^{(5)}$ $P(Z^{(5)}=A \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.3b44b$

$P(Z^{(5)}=B \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.0405$ $\therefore \hat{Z}_3^{(5)} = C$

$P(Z^{(5)}=C \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.5955$

$Z_3^{(6)}$ $P(Z^{(6)}=A \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.5961$

$P(Z^{(6)}=B \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.001$ $\therefore \hat{Z}_3^{(6)} = A$

$P(Z^{(6)}=C \mid \bar{x}_3; \bar{\theta}^{(1)}) = 0.3965$

- we can find that Z assignment does not change
  $\Rightarrow$ $\bar{\theta}$ remains unchanges

$\therefore \bar{\theta}^{(3)} = [0.15, 0.25, 0.55]$

$\bar{Z}_3 = [A\ B\ C\ B\ C\ A]$

1.3. (a) · to find θ_k that maximize $\theta_k^{(t+1)}$

$$\therefore \frac{d}{d\theta_k}\theta_k^{(t+1)} = \sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)}) \times \left(s_i \cdot \theta_k^{-1} - (1-s_i)(1-\theta_k)^{-1}\right) = 0$$

$$\sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)}) \cdot (s_i - \theta_k) = 0$$

$$\sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)}) \cdot s_i = \sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)}) \cdot \hat{\theta}_k$$

$$\therefore \theta_k^{(t+1)} = \hat{\theta}_k = \frac{\sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)}) \cdot s_i}{\sum_{i=1}^{N} p(Z^{(i)}=k \mid \bar{x}^{(i)}; \bar{\theta}^{(t)})}$$

(b) · The update is fraction which

=> numerator is sum of the number of sunny time the probality of being this city. we can understand this as an "advanced" sum of sunny days altogether considering all sample.

=> denominator is same type as numerator except that we are now considering total # of days using weight of each city on each trial.

· The equation is just updating by "expected" ratio of sunny days on total days for each city which make huge sense to me.

· Yes. correspond to my intuition. reasonable !

2. (a) · posterior $P(Z=1 \mid w_1, w_2) = \frac{P(Z=1, w_1, w_2)}{P(w_1, w_2)}$

$$\Rightarrow P(Z, w_1, w_2) = P(Z, w_2 \mid w_1) P(w_1)$$

$$= P(w_2 \mid Z, w_1) \cdot P(Z \mid w_1) \cdot P(w_1)$$

$$(w_2 \perp\!\!\!\perp w_1 \mid Z) \quad = P(w_2 \mid Z) \cdot P(Z \mid w_1) \cdot P(w_1)$$

$$= \theta_{w_2 \mid Z} \; \phi_{Z \mid w_1} \cdot P(w_1)$$

$$\Rightarrow P(w_1, w_2) = \sum_{Z=1}^{K} P(Z, w_1, w_2)$$

$$\therefore P(Z=1 \mid w_1 = \text{"machine"}, w_2 = \text{"learning"})$$

$$= \frac{\theta_{w_2=\text{"learning"} \mid Z=1} \; \phi_{Z=1 \mid w_1=\text{"machine"}} \cdot P(w_1)}{\sum_{Z=1}^{K} \theta_{w_2=\text{"learning"} \mid Z} \; \phi_{Z \mid w_1=\text{"machine"}} \cdot P(w_1)}$$

$$= \frac{\theta_{w_2=\text{"learning"} \mid Z=1} \; \phi_{Z=1 \mid w_1=\text{"machine"}}}{\sum_{Z=1}^{K} \theta_{w_2=\text{"learning"} \mid Z} \; \phi_{Z \mid w_1=\text{"machine"}}}$$

(b) $P(w_2, w_3 \cdots w_T \mid w_1) = \frac{P(w_1, w_2 \cdots w_T)}{P(w_1)}$

$$= \frac{P(w_1) \cdot P(w_2 \mid w_1) \cdot P(w_3 \mid w_2) \cdots P(w_T \mid w_{T-1})}{P(w_1)}$$

$$= P(w_2 \mid w_1) P(w_3 \mid w_2) \cdots P(w_T \mid w_{T-1})$$

$$\therefore \log P(w_2 \cdots w_T \mid w_1) = \sum_{i=1}^{T-1} \log P(w_{i+1} \mid w_i)$$

$$= \sum_{i=1}^{T-1} \log \sum_{Z=1}^{K} P(w_{i+1} \mid Z) P(Z \mid w_i)$$

$$= \sum_{i=1}^{T-1} \log \sum_{Z=1}^{K} \theta_{w_{i+1} \mid Z} \; \phi_{Z \mid w_i}$$

(c) · The numerator is summing probability cluster Z when a pair which the next word is w appears

- The denominator is summing probability of cluster z when pairs which the next word could be all words is in the dictionary appear, which we could understand as the prob. of cluster I given the words in dictionary. (All case)

- Intuitively, the update is doing

$$\theta_{wz} = \frac{\text{prob}(z=z \mid \text{next word is } w)}{\text{prob}(I=z \mid \text{next word can be all words in dict})}$$

which is very clear. the probability that the next word is w given cluster I.

- Yes P. correspond to my intuition

3. (a) · we have weights of each Gaussian π_i:
$\Rightarrow K-1$ parameters
· for mean of each Gaussian $\vec{\mu}^{(i)}$
$\Rightarrow K \times d$ parameters
· And 1 common variance shared σ
$\Rightarrow 1$ parameters
$\therefore$ total # $= K-1+dK+1 = (d+1)K$

(b) code only

(c) (1) · Because we define our BIC as

$$BIC = m\ln N - 2\ln \hat{L}$$

and our goal is to minimize BIC
$\Rightarrow$ higher mle will result in lower BIC
$\Rightarrow$ lower # of variable will result in lower BIC
· So I'll pick a lower BIC value.

(2) I'll choose the # of cluster according to the "elbow" rule. I'll see the graph and find out the point that decreating rate of mle becomes low. And that would be the point that our optimal number of cluster be.

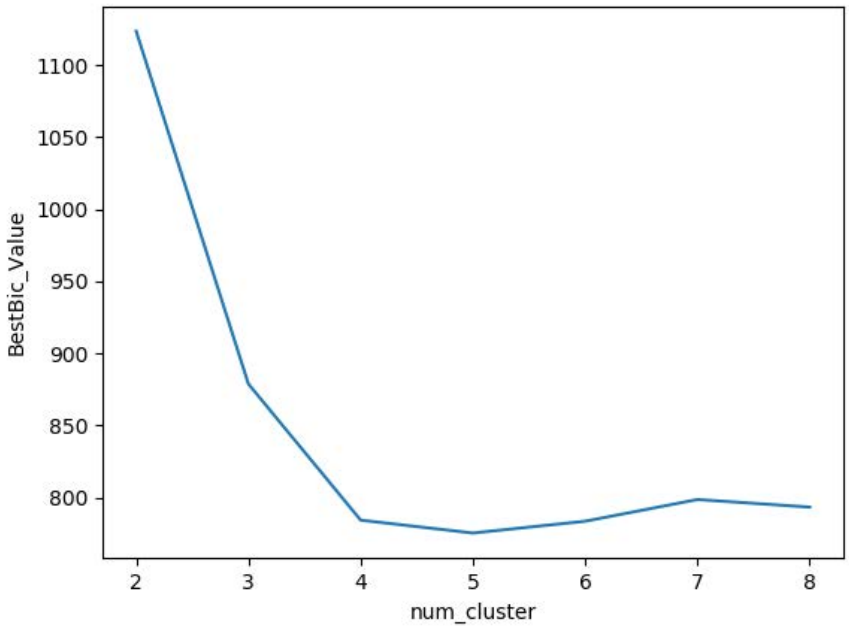
(d) · The BIC falls on range around 700 ~ 1200.
=> The curve first going down dramatically
=> Then the decreasing rate of the curve slows down
=> after k=5, the curve rise.

- Curve goes down at beginning because larger k induces a more complex model. result in higher log-likelihood => BIC goes down
- curve goes down because with k↑ the increase rate of log-likelihood becomes lower than the penalty for having more clusters.

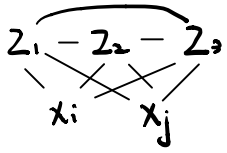
- I would choose the point k=5.
Because our BIC continue decreasing greatly until we reach the point k=5. then increasing.
=> the increasing is due to the increase of mle cannot offset the penalty of the 1 more parameter, which tells us that we should stop adding more clusters.

- Graph as below

Graph for 3.1d)



4. (a) · D-seperation for check $x_i \perp\!\!\!\perp x_j \mid \{z_1, z_2\}$ $\forall i, j$



Two possible conditions to check

=> there is obvious a path between variable we interest, so go check next

=> path $x_i \to z_3 \to x_j$ does not go through z_1 or z_2

∴ No !

∴ x_1, x_2, x_3 are not independent of each other given z_1, z_2

· d-seperation for check $z_1 \perp\!\!\!\perp z_3 \mid z_2$

$z_1 - z_2 - z_3$

Two possible condition to check

=> $\exists\ z_1 \to z_2 \to z_3$. a path between z_1 & z_3

=> ALL path between z_1, z_3 go through z_2 !

∴ Yes !

∴ $z_1 \perp\!\!\!\perp z_3 \mid z_2$

(b) $P(z_1, z_2, z_3, x_1, x_2, x_3)$

$= P(z_1) \cdot P(z_2 \mid z_1) \cdot P(z_3 \mid z_2) \cdot P(x_1 \mid z_1, z_2, z_3) \cdot P(x_2 \mid z_1, z_2, z_3)$

$\cdot P(x_3 \mid z_1, z_2, z_3)$

(c) z_1 : $2-1 = 1$; z_2 : $2 \times (2-1) = 2$

z_3 : $2 \times (2-1) = 2$

X_1, X_2, X_3 : $2 \times 2 \times 2 \times (2-1) = 8$

$\therefore$ Total # of variables $= 1 + 2 + 2 + 8 \times 3 = 29$

(d) I would use HLLM

- because as we can see in problem 3. even with the same log-likelihood, we will penalty a model more if it has relatively more parameters.
- HLLM has same log-likelihood as above model but less parameters.

$\therefore$ HLLM

5. (a) $P(S,T,R,H) = P(S)P(T|S)P(R|S)P(H|T,R)$

(b) $P(R=T|H=T) = \dfrac{P(R=T,H=T)}{P(H=T)}$

- $P(R,H) = \sum_S \sum_T P(S,T,R,H)$

$= \sum_S \sum_T P(S)P(T|S)\cdot P(R|S)\cdot P(H|T,R)$

$= \sum_S P(S)P(R|S) \sum_T P(T|S)\cdot P(H|T,R)$

$\therefore P(R=T,H=T) = [0.8\cdot 0.2\cdot(0.6\times 1 + 0.4\times 0.5)] +$

$[0.2\cdot 0.8\cdot(0.4\times 1 + 0.6\times 0.5)]$

$= 0.128 + 0.112 = 0.24$

$P(R=F,H=T) = [0.8\times 0.8\times(0.6\times 0.5 + 0.4\times 0)]$

$+ [0.2\times 0.2\times(0.4\times 0.5 + 0.6\times 0)]$

$= 0.192 + 0.008 = 0.2$

$\therefore P(H=T) = 0.24 + 0.2 = 0.44$

$\therefore P(R=T|H=T) = \dfrac{0.24}{0.44} = 0.5454$

6. (a) · if we fix all $\{\vec{u}^{(i)}\}$. then our opitimization problem becomes

$$\min_{v,\xi} \sum_{j=1}^{m} \frac{1}{2}\|\vec{v}^{(j)}\|^2 + C \sum_{i:Y_{ij}\neq 0} \xi_{ij}$$

subject to $Y_{ij}(\vec{u}^{(i)} \cdot \vec{v}^{(j)}) \geq 1 - \xi_{ij}$

$\xi_{ij} \geq 0 \quad \forall (i,j)$ where $Y_{ij} \neq 0$

· This is a single SVM problem of classification w.r.t. $\vec{v}^{(j)}$ and datapoint $\vec{u}^{(i)}$, label Y_{ij}

· Because for $\{\vec{v}^{(j)}\}$. there are totally n $\vec{v}^{(j)}$'s.

So our problem becomes n SVM problems on V

(b) · Same as (a). our optimization problem becomes:

$$\min_{v,\xi} \sum_{i=1}^{n} \frac{1}{2}\|\vec{u}^{(i)}\|^2 + C \sum_{j:Y_{ij}\neq 0} \xi_{ij}$$

subject to $Y_{ij}(\vec{u}^{(i)} \cdot \vec{v}^{(j)}) \geq 1 - \xi_{ij}$

$\xi_{ij} \geq 0 \quad \forall (i,j)$ where $Y_{ij} \neq 0$

· This is a single SVM problem of classification w.r.t. $\vec{u}^{(i)}$ and datapoint $\vec{v}^{(j)}$, label Y_{ij}

· Because for $\{\vec{u}^{(i)}\}$. there are totally n $\vec{u}^{(i)}$'s.

So our problem becomes n SVM problems on U

(c) · No.

· Because we only train on the observed datapoints, that is, train on all the (i,j) that has $Y_{ij} \neq 0$. We can also see the constrain in (a)(b) SVM. we don't care the row that $Y_{ij} = 0$. no training on it so no effect.

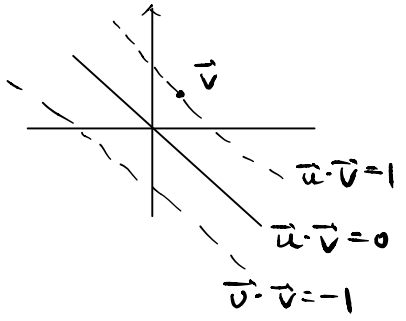
(d) · This would be a SVM problem on $\vec{u}^{(i)}$ with only one datapoint pair $\{\vec{v}^{(j)}, Y_{ij} = 1\}$

· Therefore our SVM becomes

$\min \frac{1}{2}\|\vec{u}^{(i)}\|^2 \quad \text{s.t. } \vec{u}^{(i)} \cdot \vec{v}^{(j)} \geq 1$

· Thus. $\vec{v}^{(j)}$ should be support vector

$\therefore \vec{u}^{(i)} \cdot \vec{v}^{(j)} = 1$



· To maxmargin, we would take a decision boundary that perpendicular to $\vec{v}$, so $\vec{u}$ should be paralleln to $\vec{v}$

$\Rightarrow$ suppose $\vec{u} = a \cdot \vec{v}^{(j)}$

$\therefore a \cdot \|\vec{v}^{(j)}\|^2 = 1 \Rightarrow a = 1/\|\vec{v}^{(j)}\|^2$

$\therefore \vec{u}^{(i)} = \frac{1}{\|\vec{v}^{(j)}\|^2} \cdot \vec{v}^{(j)}$

(e) i. $\vec{v}^{(1)} = [-1, 0]$; $\vec{v}^{(2)} = [0, 1]$

and we only train on $(1,1)$, $(1,2)$ and $(2,1)$

- $\vec{u}^{(1)}$ $\min \|\vec{u}^{(1)}\|^2$ s.t. $(\vec{u}^{(1)} \cdot \vec{v}^{(i)}) \cdot Y_{i1} \geq 1$ for $i = 1, 2$

we have $\vec{v}^{(1)} = [1, 0]$ with $Y = -1$.

$\vec{v}^{(2)} = [0, 1]$ with $Y = 1$

$\Rightarrow$ say $\vec{u}^{(1)} = [u_1, u_2]$

$\therefore (\vec{u}^{(1)} \cdot \vec{v}^{(1)}) Y_{11} = -u_1 \geq 1 \Rightarrow u_1 \leq 1$

$(\vec{u}^{(1)} \cdot \vec{v}^{(2)}) Y_{12} = u_2 \geq 1$

$\Rightarrow$ to minimize $\|\vec{u}^{(1)}\|^2 = u_1^2 + u_2^2$

because we have $u_1^2 \geq 1$, $u_2^2 \geq 1$

$\therefore$ we have to take $u_1 = -1$, $u_2 = 1$

$\Rightarrow \vec{u}^{(1)} = [-1, 1]$

- $\vec{u}^{(2)}$ $\min \|\vec{u}^{(1)}\|^2$ s.t. $(\vec{u}^{(2)} \cdot \vec{v}^{(i)}) \cdot Y_{i2} \geq 1$

we have $\vec{v}^{(1)} = [1, 0]$, $Y = 1$

$\Rightarrow$ say $\vec{u}^{(2)} = [u_1, u_2]$

$\therefore (\vec{u}^{(2)} \cdot \vec{v}^{(1)}) \cdot 1 = u_1 \geq 1$

$\Rightarrow$ to minimize $\|\vec{u}^{(2)}\|^2 = u_1^2 + u_2^2$

because we have $u_1^2 \geq 1$, $u_2^2 \geq 0$

$\therefore$ we have to take $u_1 = 1$, $u_2 = 0$

$\therefore \vec{u}_2 = [1, 0]$

$\therefore U = \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}$

$\therefore Y_{22} = \vec{u}^{(2)} \cdot \vec{v}^{(2)} = [1, 0] \cdot [0, 1] = 0$

$\therefore$ we cannot predict Y_{22}

ii. $\vec{u}^{(1)} = [-1, 1]$; $\vec{u}^{(2)} = [1, 0]$

and we only train on (1,1), (1,2) and (2,1)

- $\vec{v}^{(1)}$ $\min \|\vec{v}^{(1)}\|^2$ s.t. $(\vec{u}^{(i)} \cdot \vec{v}^{(1)}) \cdot Y_{i1} \geq 1$ for $i = 1, 2$

we have $\vec{u}^{(1)} = [-1, 1]$ w.l $Y = -1$

$\vec{u}^{(2)} = [1, 0]$ w.l $Y = 1$

$\Rightarrow$ say $\vec{v}^{(1)} = [v_1, v_2]$

$\therefore (\vec{u}^{(1)} \cdot \vec{v}^{(1)}) Y_{11} = -v_1 + v_2 \geq 1$

$(\vec{u}^{(1)} \cdot \vec{v}^{(2)}) Y_{12} = v_1 \geq 1$

$\Rightarrow$ to minimize $\|\vec{v}^{(1)}\|^2 = v_1^2 + v_2^2$

because we have

$v_1^2 - 2v_1v_2 + v_2^2 \geq 1$

$v_1^2 + v_2^2 \geq 1 + 2v_1v_2$

$\because v_1 \geq 1 \Rightarrow v_2 = 0$ for max

$\therefore v_1 = 1$

$\therefore \vec{v}^{(1)} = [1, 0]$

- $\vec{u}^{(2)}$ $\min \|\vec{v}^{(1)}\|^2$ s.t. $(\vec{u}^{(2)} \cdot \vec{v}^{(1)}) \cdot Y_{21} \geq 1$

we have $\vec{u}^{(1)} = [-1, 1]$. $Y = 1$

$\Rightarrow$ say $\vec{v}^{(2)} = [v_1, v_2]$

$\therefore (\vec{u}^{(2)} \cdot \vec{v}^{(1)}) \cdot 1 = -v_1 + v_2 \geq 1$

$\Rightarrow$ to minimize $\|\vec{v}^{(2)}\|^2 = v_1^2 + v_2^2$

because we have $v_1^2 - 2v_1v_2 + v_2^2 \geq 1$

$v_1^2 + v_2^2 \geq 1 + 2v_1v_2$

$\geq 1 + 2v_1(1 + v_1)$

$\geq 1 + 2v_1 + 2v_1^2$

$\geq 2(\frac{1}{2}+v_1)^2+\frac{1}{2}$

$\therefore$ we have to have $v_1 = -\frac{1}{2} \Rightarrow v_2 \geq \frac{1}{2}$

$\therefore$ to minimize $\|\vec{v}^{(2)}\|^2 \Rightarrow v_2 = \frac{1}{2}$

$\therefore \vec{v}_2 = [-\frac{1}{2}, \frac{1}{2}]$

$\therefore Y_{22} = \vec{u}^{(2)} \cdot \vec{v}^{(2)} = [1,0] \cdot [-\frac{1}{2}, \frac{1}{2}] = -\frac{1}{2}$

< 0

$\therefore \hat{Y}_{22} = -1$. we can predict it.

APPENDIX

1.2 (b)

In [1]:
```
import numpy as py
```

In [2]:
```
def calc_z(p_a, p_b, p_c, sunny):
    x_a = ((p_a) ** sunny) * (1 - p_a) ** (10 - sunny)
    x_b = ((p_b) ** sunny) * (1 - p_b) ** (10 - sunny)
    x_c = ((p_c) ** sunny) * (1 - p_c) ** (10 - sunny)
    sum_abc = x_a + x_b + x_c
    return (x_a/sum_abc, x_b/sum_abc, x_c/sum_abc)
```

In [14]:
```
calc_z(0.75,0.25,0.55,8)
```

Out[14]:
```
(0.7859387940650213, 0.0010781053416529785, 0.21298310059332584)
```

In [15]:
```
calc_z(0.75,0.25,0.55,3)
```

Out[15]:
```
(0.009421150914142334, 0.7631132240455292, 0.2274656250403285)
```

In [16]:
```
calc_z(0.75,0.25,0.55,5)
```

Out[16]:
```
(0.1664595841967641, 0.1664595841967641, 0.6670808316064718)
```

In [17]:
```
calc_z(0.75,0.25,0.55,2)
```

Out[17]:
```
(0.0012670049499898517, 0.9236466085426018, 0.07508638650740826)
```

In [18]:
```
calc_z(0.75,0.25,0.55,6)
```

Out[18]:
```
(0.36446047786379454, 0.04049560865153273, 0.5950439134846727)
```

In [19]:
```
calc_z(0.75,0.25,0.55,7)
```

Out[19]:
```
(0.5961224206794469, 0.007359536057770949, 0.39651804326278217)
```

3 (b)

```
for iter in range(0,num_iter):
    """
        E-Step
        In the first step, we find the expected log-likelihood of the data which is equivalent to:
        finding cluster assignments for each point probabilistically
        In this section, you will calculate the values of zk(n,k) for all n and k according to current valu
    """
    # TODO

    for i in range(0, N):
        for j in range(0, num_K):
            zk[i, j] = np.log(pk[j]) + calc_logpdf(trainX[i], mu[j], si2)
        zk[i] = zk[i] - logsumexp(zk[i])


    """
        M-step
        Compute the GMM parameters from the expressions which you have in your writeup
    """

    # Estimate new value of pk
    # TODO
    for j in range(0, num_K): pk[j] = np.exp(logsumexp(zk[:,j])) / N

    # Estimate new value for means
    # TODO
    for j in range(0, num_K): mu[j] = np.matmul([np.exp(zk[:,j])], trainX) / np.exp(logsumexp(zk[:,j]))

    # Estimate new value for sigma^2
    # TODO
    transform = np.zeros([N, num_K])
    for i in range(0, N):
        for j in range(0, num_K):
            transform[i, j] = (trainX[i] - mu[j]).dot(trainX[i] - mu[j])
    si2 = np.exp(logsumexp(zk, b = transform)) / (N * D)

# Computing the expected likelihood of data for the optimal parameters computed
# TODO
for i in range(0, N):
    for j in range(0, num_K):
        zk[i, j] = np.log(pk[j]) + calc_logpdf(trainX[i], mu[j], si2)

log_like = np.sum(logsumexp(zk, 1))

# Compute the BIC for the current cluster
# TODO
BIC = (D + 1) * num_K * np.log(N) - 2 * log_like
```

3 (c) (d)

```
print("We'll try different numbers of clusters with GMM, using multiple runs for each to identify the 'best' results")
trainX = get_data()
num_K = range(2, 9) # List of cluster sizes
BIC_K = np.zeros(len(num_K))
means = {} # Dictionary mapping cluster size to corresponding matrix of means
cluster_proportions = {} # Dictionary mapping cluster size to corresponding mixture proportions vector
z_K = {}
sigma2 = {} # Dictionary mapping cluster size to the learned variance value
for idx in range(len(num_K)):
    # Running
    k = num_K[idx]
    print("%d clusters..." % k)
    bestBIC = float("inf")
    for i in range(1, 11):
        # TODO: Run gmm function 10 times and get the best
        # set of parameters for this particular value of k
        log_like = gmm(trainX, k)[4]
        if log_like < bestBIC: bestBIC = log_like
    BIC_K[idx] = bestBIC

# TODO: Part d: Make a plot to show BIC as function of clusters K
plt.plot(num_K, BIC_K)
plt.xlabel("num_cluster")
plt.ylabel("BestBic_Value")
plt.show()
```