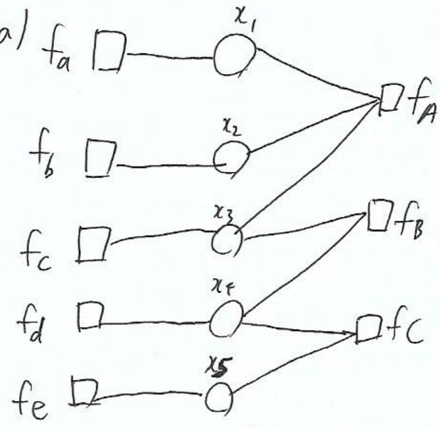
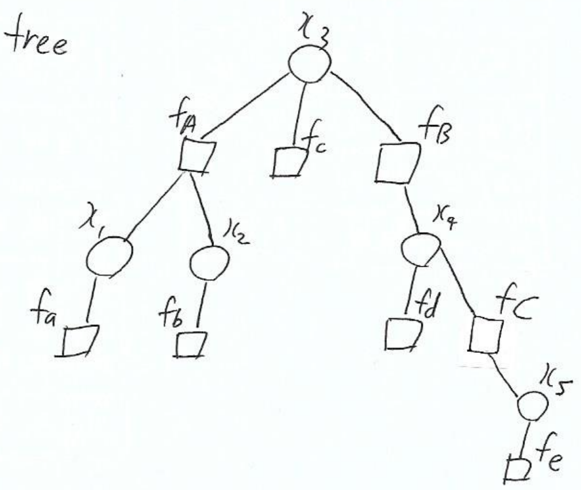
ML Assn 4 20160963 성하빈

1. (a)



tree



(b)

$\mu_{a\to 1} = f_a(x_1)$

$\mu_{b\to 2} = f_b(x_2)$

$\mu_{c\to 3} = f_c(x_3)$

$\mu_{d\to 4} = f_d(x_4)$

$\mu_{e\to 5} = f_e(x_5)$

$\mu_{1\to A} = \mu_{a\to 1} = f_a(x_1)$

$\mu_{2\to A} = \mu_{b\to 2} = f_b(x_2)$

$$\mu_{A\to 3} = \max_{x_1,x_2}\left\{f_A(x_1,x_2,x_3)\cdot\mu_{1\to A}\cdot\mu_{2\to A}\right\}$$
$$= \max_{x_1,x_2}\left\{f_A(x_1,x_2,x_3)\cdot f_a(x_1)\cdot f_b(x_2)\right\}$$

$\mu_{5\to C} = \mu_{e\to 5} = f_e(x_5)$

$$\mu_{C\to 4} = \max_{x_5}\left\{f_C(x_4,x_5)\cdot\mu_{5\to C}\right\} = \max_{x_5}\left\{f_C(x_4,x_5)\cdot f_e(x_5)\right\}$$

$$\mu_{4\to B} = \mu_{d\to 4}\cdot\mu_{C\to 4} = f_d(x_4)\cdot\max_{x_5}\left\{f_C(x_4,x_5)\cdot f_e(x_5)\right\}$$

$$\mu_{B\to 3} = \max_{x_4}\left\{f_B(x_3,x_4)\cdot\mu_{4\to B}\right\} = \max_{x_4}\left\{f_B(x_3,x_4)\cdot f_d(x_4)\cdot\max_{x_5}\left\{f_C(x_4,x_5)\cdot f_e(x_5)\right\}\right\}$$

$$P_{max} = \max_{x_3}\left\{\mu_{A\to 3}\cdot\mu_{B\to 3}\cdot\mu_{c\to 3}\right\}$$

$$= \max_{x_3}\Big\{f_c(x_3)\cdot\underbrace{\max_{x_1,x_2}\left\{f_A(x_1,x_2,x_3)\cdot f_a(x_1)\cdot f_b(x_2)\right\}}_{e^2 :\, x_1=0,\ x_2=1}\cdot\underbrace{\max_{x_4}\left\{f_B(x_3,x_4)\cdot f_d(x_4)\cdot\max_{x_5}\left\{f_C(x_4,x_5)\, f_e(x_5)\right\}\right\}}_{e^2}\Big\}$$

$x_3=0$: $1\cdot\boxed{\max_{x_1,x_2}\left\{f_A\cdot e^{x_1+2x_2}\right\}}\ \boxed{\max_{x_4}\left\{f_B\cdot e^{2x_4}\max_{x_5}\left\{f_C\cdot e^{2x_5}\right\}\right\}}$: e^4

$x_3=1$: $e^3\cdot\boxed{\max_{x_1,x_2}\left\{f_A\cdot e^{x_1+2x_2}\right\}}\cdot\boxed{\max_{x_4}\left\{f_B\cdot e^{2x_4}\max_{x_5}\left\{f_C\cdot e^{2x_5}\right\}\right\}}\cdot e^5$

$1 :\ x_1=0,\ x_2=0$

$e^2 :\ x_4=0,\ x_5=1$

max

$\to$ $x_1=0$, $x_2=0$, $x_3=1$, $x_4=0$, $x_5=1$ — most likely configuration

2. (a)

x	0	1
$p(X_1=x)$	0.34	0.66
$p(X_2=x)$	0.45	0.55
$p(X_3=x)$	0.45	0.55
$p(X_4=x)$	0.46	0.54

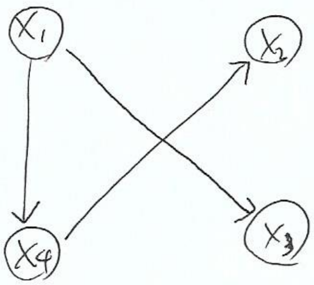
(x,y)	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$
$p(X_1=x, X_2=y)$	0.14	0.20	0.31	0.35
$p(X_1=x, X_3=y)$	0.13	0.21	0.32	0.34
$p(X_1=x, X_4=y)$	0.12	0.22	0.34	0.32
$p(X_2=x, X_3=y)$	0.19	0.26	0.26	0.29
$p(X_2=x, X_4=y)$	0.18	0.27	0.28	0.27
$p(X_3=x, X_4=y)$	0.21	0.24	0.25	0.30

(b)

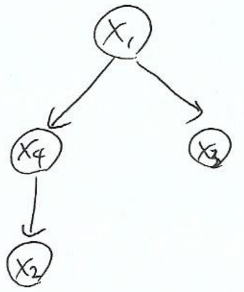
$I(X_1, X_2)$	0.001526	
$I(X_1, X_3)$	0.004795	3rd
$I(X_1, X_4)$	0.012025	1st
$I(X_2, X_3)$	0.001276	
$I(X_2, X_4)$	0.005948	2nd
$I(X_3, X_4)$	0.000073	

$$I(X_1, X_2) = 0.14 \log \frac{0.14}{0.34 \times 0.45} + 0.2 \log \frac{0.2}{0.34 \times 0.55} + 0.31 \log \frac{0.31}{0.66 \times 0.45} + 0.35 \log \frac{0.35}{0.66 \times 0.55}$$

(c)

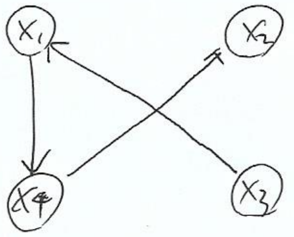


:

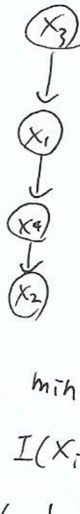


$$P_{T_1}(x) = p(x_1) \cdot p(x_3 | x_1) \cdot p(x_4 | x_1) \cdot p(x_2 | x_4)$$

(d)



:



$$P_{T_3}(x) = p(x_3) \cdot p(x_1 | x_3) \cdot p(x_4 | x_1) \cdot p(x_2 | x_4)$$

(e) The Chow-Liu Algorithm is equivalent to minimizing KL divergence. Both T_1 and T_3 have same weights (direction doesn't matter in $I(X_i, X_j)$), and same joint distribution because $p(x_1) \cdot p(x_3|x_1) = p(x_3) p(x_1|x_3) = p(x_1, x_3)$. We know $P_{T_1} = P_{T_3}$, so $KL(p \| P_{T_1}) - KL(p \| P_{T_3}) = 0$.

3. (a)

$$r_{ik} = \begin{cases} 1 & \text{if } k = \arg\min_k \|x_i - \mu_k\|_2^2 \\ 0 & \text{otherwise} \end{cases}$$

we should find an assignment that minimizes the cost.

(b) $$\mu_k = \frac{\sum_i r_{ik} x_i}{\sum_i r_{ik}}$$

take the gradient of cost function with respect to μ and set it to 0.

$$\frac{\partial L}{\partial \mu} = \frac{\partial}{\partial \mu}\left(\sum_i \sum_k \frac{1}{2} r_{ik} \|x^{(i)} - \mu_k\|_2^2 \right) = \sum_i r_{ik}(x_i - \mu_k) = \sum_i r_{ik} x_i - \mu_k \sum_i r_{ik} = 0$$

$$\mu_k = \frac{\sum_i r_{ik} x_i}{\sum_i r_{ik}}$$

(c) Start with randomly chosen k centroids $\{\mu_k\}$.

Assignment: given μ. calculate r_{ik} as (3a)

Update: given r, calculate μ_k as (3b)

Repeat Assignment-Update until convergence: r and μ does not change

L_t is monotonically decreasing in t. In Assignment Step, each point is assigned to the lowest cost centroid, so L decreases. In Update step, we take the gradient of cost and set it to 0, which means the new centroid is the centroid that L is minimum. So each step makes L non-increase (decrease), so L is monotonically decreasing: $L_t \geq L_{t+1}$ for every $t \geq 1$.
The lower bound of L is 0 since $r_{ik} \|x^{(i)} - \mu_k\|_2^2 \geq 0$. Due to monotone convergence theorem, this sequence has a finite limit, thus converges. But there is no guarantee that it converges to global optimality.

(d) dist = 0.5 * torch.norm(x-ctmp, dim=1)**2

after 2 updates the algorithm converges to 4.559995.

obtained centers: (1.9163, -1.9143), (-2.0952, 2.0540)

4. (a)

$$-\sum_x \log\left(\frac{1}{1+\exp(w^T x)}\right) - \sum_z \log\left(1-\frac{1}{1+\exp(w^T G_\theta(z))}\right) + \frac{C}{2}\|w\|_2^2$$

$$= +\sum_x \log(1+\exp(w^T x)) + \sum_z \log(1+\exp(w^T G_\theta(z))) - \sum_z w^T G_\theta(z) + \frac{C}{2}\|w\|_2^2$$

(b)

$$H(A) = \begin{bmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ 0 & 0 & C \cdots & 0 \\ & & \vdots & \\ 0 & 0 & 0 \cdots & C \end{bmatrix}$$

for any $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$, $x^T H(A) x = [x_1 \cdots x_n] \begin{bmatrix} C & 0 & \cdots & 0 \\ 0 & C & \cdots & 0 \\ & & \vdots & \\ 0 & 0 & \cdots & C \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = Cx_1^2 + Cx_2^2 + \cdots + Cx_n^2 \geq 0$

because $(x_1^2 + \cdots x_n^2) \geq 0$ and hyperparameter $C \geq 0$ given by condition.

thus $H(A)$ is positive semi-definite, and using Fact1, (A) is convex.

(c)

$$H(B) = \begin{bmatrix} \frac{b_1^2 \exp(w^T b)}{(1+\exp(w^T b))^2} & \frac{b_1 b_2 \exp(w^T b)}{(1+\exp(w^T b))^2} & \cdots & \frac{b_1 b_n \exp(w^T b)}{(1+\exp(w^T b))^2} \\ \frac{b_1 b_2 \exp(w^T b)}{(1+\exp(w^T b))^2} & \cdots & \cdots & \\ \vdots & & & \\ \frac{b_1 b_n \exp(w^T b)}{(1+\exp(w^T b))^2} & \cdots & \cdots & \frac{b_n^2 \exp(w^T b)}{(1+\exp(w^T b))^2} \end{bmatrix} \quad \text{when } b = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix}$$

for any $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$, $x^T H(B) x = \frac{\exp(w^T b)}{(1+\exp(w^T b))^2}(b_1 x_1 + b_2 x_2 + \cdots b_n x_n)^2 \geq 0$

thus $H(B)$ is positive semi-definite, and using Fact1, (B) is convex.

(d)

$$\underbrace{\sum_x \log(1+\exp(w^T x))}_{(B)} + \underbrace{\sum_z \log(1+\exp(w^T G_\theta(z)))}_{(B)} \underbrace{- \sum_z w^T G_\theta(z) + \frac{C}{2}\|w\|_2^2}_{(A)}$$

using fact 2, the cost is convex.

(e)

$$\text{Lagrangian} = \sum_x \log(1+\exp(\xi_x)) + \sum_z \log(1+\exp(\xi_z)) - \sum_z w^T G_\theta(z) + \frac{C}{2}\|w\|_2^2$$
$$+ \lambda_x(\xi_x - w^T x) + \lambda_z(\xi_z - w^T G_\theta(z))$$

(f) $\frac{C}{2}\|w\|_2^2 - w^T b$ is convex, so $\nabla_w\left(\frac{C}{2}\|w\|_2^2 - w^T b\right) = Cw - b = 0$, $w = \frac{1}{C} b$

(g) $\lambda\xi + \log(1+\exp\xi)$ is convex, so $\nabla_\xi(\lambda\xi + \log(1+\exp\xi)) = 0$

$$\lambda + \frac{\exp(\xi)}{1+\exp(\xi)} = 0, \qquad \xi = \log\left(\frac{-\lambda}{1+\lambda}\right)$$

(h)

$$L = \boxed{\lambda_x\xi_x + \sum_x \log(1+\exp\xi_x)} + \boxed{\lambda_z\xi_z + \sum_z \log(1+\exp\xi_z)} + \boxed{\frac{C}{2}\|w\|_2^2 - w^T\left(\sum_z G_\theta(z) + \lambda_x x + \lambda_z G_\theta(z)\right)}$$

$$\xi_x = \log\left(\frac{-\lambda_x}{1+\lambda_x}\right) \qquad \xi_z = \log\left(\frac{-\lambda_z}{1+\lambda_z}\right) \qquad w = \frac{1}{C}\left(\sum_z G_\theta(z) + \lambda_x x + \lambda_z G_\theta(z)\right)$$

$$g(\lambda) = \lambda_x \log\left(\frac{-\lambda_x}{1+\lambda_x}\right) - \sum_x \log(1+\lambda_x) + \lambda_z \log\left(\frac{-\lambda_z}{1+\lambda_z}\right) - \sum_z \log(1+\lambda_z) - \frac{1}{2C}\left(\sum_z G_\theta(z) + \lambda_x x + \lambda_z G_\theta(z)\right)$$

$$\max_\theta \min_w \sum_x \log(1+\exp(w^T x)) + \sum_z \log(1+\exp(w^T G_\theta(z))) - \sum_z w^T G_\theta(z) + \frac{C}{2}\|w\|_2^2$$

$$\Leftrightarrow \max_\theta \max_\lambda \lambda_x \log\left(\frac{-\lambda_x}{1+\lambda_x}\right) - \sum_x \log(1+\lambda_x) + \lambda_z \log\left(\frac{-\lambda_z}{1+\lambda_z}\right) - \sum_z \log(1+\lambda_z) - \frac{1}{2C}\left(\sum_z G_\theta(z) + \lambda_x x + \lambda_z G_\theta(z)\right)$$

(i) loss = criterion (logit, target1)

loss = criterion (logit, target2)

```python
for iter in range(10):
    #[Your task]
    ##############################
    ## compute the distance between points and cluster centers
    ## Dimensions: dist (2x20)
    dist = 0.5*torch.norm(x - ctmp, dim=1) ** 2
    ##############################
```

```
    #[Your task1]
    ##############################
    ## implement the discriminator loss (-logD trick)
    loss = criterion(logit, target1)
    ##############################

    print("E: %d; B: %d; DLoss: %f" % (epoch,batch_idx,loss.ite
    loss.backward()
    dopt.step()
    dopt.zero_grad()

gopt.zero_grad()
for k in range(genIter):
    xhat = gen(z)
    logit = disc(xhat)
    #[Your task2]
    ##############################
    ## implement the generator loss (-logD trick)
    loss = criterion(logit, target2)
    ##############################
    loss.backward()
```