CSE250A HW8

Jiping Lin A15058075

1.(a) See Code

(b) $P(\{R_j=r_j^{(t)}\}_{j\in\Omega_t}) = \sum_{i=1}^{k} P(Z=i, \{R_j=r_j^{(t)}\}_{j\in\Omega_t})$ 「Marg.」

$= \sum_{i=1}^{k} P(Z=i)\, P(\{R_j=r_j^{(t)}\}_{j\in\Omega_t} \mid Z=i)$ 「PR」

$= \sum_{i=1}^{k} P(Z=i) \prod_{j\in\Omega_t} P(R_j=r_j^{(t)} \mid Z=i)$ 「CI」

(c) $$P(Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t}) = \frac{P(Z=i, \{R_j=r_j^{(t)}\}_{j\in\Omega_t})}{\sum_{i'=1}^{k} P(Z=i', \{R_j=r_j^{(t)}\}_{j\in\Omega_t})}$$ 「By Baye's Rule and denom. is Marg. $(= P(\{R_j=r_j^{(t)}\}_{j\in\Omega_t})$」

$$= \frac{P(Z=i) \prod_{j\in\Omega_t} P(R_j=r_j^{(t)} \mid Z=i)}{\sum_{i'=1}^{k} P(Z=i') \prod_{j\in\Omega_t} P(R_j=r_j^{(t)} \mid Z=i')}$$ 「pt.(b) w/o Σ」

(d) Root: $P(Z=i) = \frac{1}{T}\sum_{t=1}^{T} P(Z=i \mid V_t=v_t)$ ↳ visible

$= \frac{1}{T}\sum_{t=1}^{T} P(Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t})$

$= \frac{1}{T}\sum_{t=1}^{T} \rho_{it}$

Other: $$P(R_j=1 \mid Z=i) = \frac{\sum_{t=1}^{T} P(R_j=1, Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t})}{\sum_{t=1}^{T} P(Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t})}$$

$$= \frac{\sum_{\{t \mid j\in\Omega_t\}} I(R_j^{(t)}, 1)\, P(Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t}) + \sum_{\{t \mid j\notin\Omega_t\}} P(Z=i \mid \{R_j=r_j^{(t)}\}_{j\in\Omega_t}) \cdot P(R_j=1 \mid Z=i, \{R_j=r_j^{(t)}\}_{j\in\Omega_t})}{\sum_{t=1}^{T} \rho_{it}}$$

「Split t into t, s.t. $j\in\Omega_t$ so we can use indicator & t, s.t. $j\notin\Omega_t$ where we use PR」

$$= \frac{\sum_{\{t \mid j\in\Omega_t\}} I(R_j^{(t)}, 1)\, \rho_{it} + \sum_{\{t \mid j\notin\Omega_t\}} \rho_{it}\, P(R_j=1 \mid Z=i)}{\sum_{t=1}^{T} \rho_{it}}$$ 「CI, since in this sum, j is not in Ω_t」

(e) likelihood increases, the form is

iter	$\mathcal{L}$
0	-27.0358
1	-17.5604
2	-16.0024
4	-15.0606
8	-14.5016
16	-14.2638
32	-14.1802
64	-14.1701

iter	$\mathcal{L}$
128	-14.1640
256	-14.1637

(f) See code

The rec. seems fit for my taste for first few movies.

2.(a) $P(y=1|\vec{x}) = \dfrac{P(\vec{x}|y=1)P(y=1)}{\sum_j P(\vec{x}, y=j)}$ ⌜BR & Marg.⌟

$$= \frac{P(\vec{x}|y=1)P(y=1)}{P(\vec{x}|y=0)P(y=0)+P(\vec{x}|y=1)P(y=1)}$$

$$= \frac{(2\pi)^{-d/2}|\Sigma_1|^{-1/2}\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_1)^T\Sigma_1^{-1}(\vec{x}-\vec{\mu}_1))\pi_1}{(2\pi)^{-d/2}|\Sigma_0|^{-1/2}\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_0)^T\Sigma_0^{-1}(\vec{x}-\vec{\mu}_0))\pi_0 + \text{num.}}$$

(b) In denom. of part (a), 2 parts now share the term $(2\pi)^{-d/2}|\Sigma|^{-1/2}$, so this cancels out w/ the same part in num., so we have

$$P(y=1|\vec{x}) = \frac{\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_1)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_1))\pi_1}{\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_0)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_0))\pi_0+\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_1)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_1))\pi_1}$$

so $\dfrac{1}{P(y=1|\vec{x})} = 1 + \dfrac{\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_0)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_0))\pi_0}{\exp(-\frac{1}{2}(\vec{x}-\vec{\mu}_1)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_1))\pi_1}$

$= 1+\exp[-\frac{1}{2}((\vec{x}-\vec{\mu}_0)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_0)-(\vec{x}-\vec{\mu}_1)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_1)]\cdot\exp(\log\frac{\pi_0}{\pi_1})$ ⌜$\frac{\pi_0}{\pi_1}=e^{\log\frac{\pi_0}{\pi_1}}$⌟

let $-z = -\frac{1}{2}((\vec{x}-\vec{\mu}_0)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_0)-(\vec{x}-\vec{\mu}_1)^T\Sigma^{-1}(\vec{x}-\vec{\mu}_1))+\log\frac{\pi_0}{\pi_1}$

$= -\frac{1}{2}((\vec{x}^T\Sigma^{-1}\vec{x}-\vec{x}^T\Sigma^{-1}\vec{\mu}_0-\vec{\mu}_0^T\Sigma^{-1}\vec{x}+\vec{\mu}_0^T\Sigma^{-1}\vec{\mu}_0)$
$\quad -(\vec{x}^T\Sigma^{-1}\vec{x}-\vec{x}^T\Sigma^{-1}\vec{\mu}_1-\vec{\mu}_1^T\Sigma^{-1}\vec{x}+\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1))+\log\frac{\pi_0}{\pi_1}$

$= -\frac{1}{2}((\underbrace{\vec{x}^T\Sigma^{-1}\vec{x}}_{\text{cancel}}-2\vec{x}^T\Sigma^{-1}\vec{\mu}_0+\vec{\mu}_0^T\Sigma^{-1}\vec{\mu}_0)$
$\quad -(\underbrace{\vec{x}^T\Sigma^{-1}\vec{x}}_{\text{cancel}}-2\vec{x}^T\Sigma^{-1}\vec{\mu}_1+\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1))+\log\frac{\pi_0}{\pi_1}$

$= -\frac{1}{2}(\vec{x}^T(2\Sigma^{-1}(\vec{\mu}_1-\vec{\mu}_0)+\vec{\mu}_0^T\Sigma^{-1}\vec{\mu}_0-\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1)+\log\frac{\pi_0}{\pi_1}$

$= \vec{x}^T\Sigma^{-1}(\vec{\mu}_0-\vec{\mu}_1)-\frac{1}{2}(\vec{\mu}_0^T\Sigma^{-1}\vec{\mu}_0-\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1)+\log\frac{\pi_0}{\pi_1}$

⌜Since $\vec{\mu}_i^T\Sigma^{-1}\vec{x}$ is scalar and Σ^{-1} is positive definite, i.e. $(\Sigma^{-1})^T=\Sigma^{-1}$, then $(\vec{\mu}_i^T\Sigma^{-1}\vec{x})^T=\vec{\mu}_i^T\Sigma^{-1}\vec{x}=\vec{x}^T\Sigma^{-1}\vec{\mu}_i$⌟

Since $\Sigma^{-1}(\vec{\mu}_0-\vec{\mu}_1)$ is col. vector
then $\vec{x}^T\Sigma^{-1}(\vec{\mu}_0-\vec{\mu}_1)$ is equivalent to $\vec{x}\cdot(\Sigma^{-1}(\vec{\mu}_0-\vec{\mu}_1))$
then writing z as $\vec{w}\cdot\vec{x}+b$, we have ($-z$ before, flip sign)
$\vec{w}=\Sigma^{-1}(\vec{\mu}_1-\vec{\mu}_0)$, $b=-\log\frac{\pi_0}{\pi_1}+\frac{1}{2}(\vec{\mu}_0^T\Sigma^{-1}\vec{\mu}_0-\vec{\mu}_1^T\Sigma^{-1}\vec{\mu}_1)$

(c) $\dfrac{P(y=1|\vec{x})}{P(y=0|\vec{x})} = \dfrac{P(y=1|\vec{x})}{1-P(y=1|\vec{x})} = \dfrac{1/(1+e^{-z})}{1-1/(1+e^{-z})} = \dfrac{1/(1+e^{-z})}{e^{-z}/(1+e^{-z})} = \dfrac{1}{e^{-z}}$

then if this fraction is constant, then $\frac{1}{e^{-z}}=k$, which means z must be constant, then $\vec{w}\cdot\vec{x}+b=z=-\log\frac{1}{k}$, which is indeed constant.
if $k=1$, then $z=-\log 1=0 \Rightarrow \vec{w}\cdot\vec{x}+b=0$.

3. (a) $\mathcal{L}(\bar{v}) = \sum_{t=1}^{T} \log P(y_t|\bar{x}_t)$

$$= \sum_{t=1}^{T} y_t \log P(y_t=1|\bar{x}_t) + (1-y_t)\log P(y_t=0|\bar{x}_t)$$

$$= \sum_{t=1}^{T} y_t \log(1-e^{-\bar{v}\cdot\bar{x}_t}) + (1-y_t)\log e^{-\bar{v}\cdot\bar{x}_t}$$

$$= \sum_{t=1}^{T} y_t \log(1-e^{-\bar{v}\cdot\bar{x}_t}) - (1-y_t)(\bar{v}\cdot\bar{x}_t)$$

(b) $\frac{\partial \mathcal{L}}{\partial v} = \sum_{t=1}^{T} y_t \frac{-(-\bar{x}_t e^{-\bar{v}\cdot\bar{x}_t})}{1-e^{-\bar{v}\cdot\bar{x}_t}} - (1-y_t)$

$$= \sum_{t=1}^{T} y_t \bar{x}_t \frac{e^{-\bar{v}\cdot\bar{x}_t}}{1-e^{-\bar{v}\cdot\bar{x}_t}} - (1-y_t)\bar{x}_t$$

$$= \sum_{t=1}^{T} \left(\frac{y_t e^{-\bar{v}\cdot\bar{x}_t}}{1-e^{-\bar{v}\cdot\bar{x}_t}} + y_t - 1\right)\bar{x}_t$$

$$= \sum_{t=1}^{T} \left(\frac{y_t e^{-\bar{v}\cdot\bar{x}_t} + y_t - y_t e^{-\bar{v}\cdot\bar{x}_t} - 1 + e^{-\bar{v}\cdot\bar{x}_t}}{1-e^{-\bar{v}\cdot\bar{x}_t}}\right)\bar{x}_t$$

$$= \sum_{t=1}^{T} \left(\frac{y_t - (1-e^{-\bar{v}\cdot\bar{x}_t})}{1-e^{-\bar{v}\cdot\bar{x}_t}}\right)\bar{x}_t = \sum_{t=1}^{T}\left(\frac{y_t - \rho_t}{\rho_t}\right)\bar{x}_t, \quad \rho_t = 1-e^{-\bar{v}\cdot\bar{x}_t}$$

$$= P(y_t=1|\bar{x}_t)$$

(ii)

(c) $P(y=1|\bar{x}) = 1-e^{-\bar{v}\cdot\bar{x}}$

$$= 1-e^{-\sum v_i x_i}$$

$$= 1-e^{\sum \log(1-p_i)^{x_i}}$$

$$= 1-\prod e^{\log(1-p_i)^{x_i}}$$

$$= 1-\prod(1-p_i)^{x_i},$$

$P(y=0|\bar{x}) = 1-P(y=1|\bar{x})$

$$= \prod(1-p_i)^{x_i}$$

So this is equivalent to noisy-OR

(d) $\frac{\partial \mathcal{L}}{\partial p_i} = \frac{\partial \mathcal{L}}{\partial v_i}\cdot\frac{\partial v_i}{\partial p_i} = \left(\frac{1}{1-p_i}\right)\frac{\partial L}{\partial v_i}$

「Chain rule $v = f(p)$, $\mathcal{L} = g(v)$,

$$\Rightarrow \mathcal{L}'(p) = \mathcal{L}'(f(p))\,v'(p)$$

$$= \mathcal{L}'(v)\,v'(p)」$$

(e) GA: $p_i + \eta\left(\frac{\partial \mathcal{L}}{\partial p_i}\right) = p_i + \frac{p_i(1-p_i)}{T_i}\left(\frac{1}{1-p_i}\right)\frac{\partial \mathcal{L}}{\partial v_i}$

$$= p_i + \frac{p_i}{T_i}\sum_{t=1}^{T}\left(\frac{y_t-\rho_t}{\rho_t}x_{it}\right), \text{ since it is ith term of } \bar{x}_t$$

$$= p_i + \frac{p_i}{T_i}\sum_{t=1}^{T}\left(\frac{y_t x_{it}}{\rho_t} - x_{it}\right) = p_i + \frac{p_i}{T_i}\left(\sum_{t=1}^{T}\frac{y_t x_{it}}{\rho_t} - \sum_{t=1}^{T} x_{it}\right) = p_i + \frac{p_i}{T_i}\left(\sum_{t=1}^{T}\frac{y_t x_{it}}{\rho_t} - T_i\right)$$

$$= \frac{p_i}{T_i}\sum_{t=1}^{T}\frac{y_t x_{it}}{\rho_t}$$

4.(a) $P(y=1, y'=1 \mid \bar{x}, \bar{x}', s=1)$

$$= \frac{\underbrace{P(s=1 \mid y=1, y'=1, \bar{x}, \bar{x}')}_{(=1 \text{ since } y=y')} P(y=1, y'=1 \mid \bar{x}, \bar{x}')}{P(s=1 \mid \bar{x}, \bar{x}')} \quad \ulcorner BR \lrcorner$$

$$= \frac{P(y=1 \mid \bar{x}, \bar{x}') P(y'=1 \mid y=1, \bar{x}, \bar{x}')}{P(s=1 \mid \bar{x}, \bar{x}')} \quad \ulcorner PR \lrcorner$$

$$= \frac{P(y=1 \mid \bar{x}) P(y'=1 \mid \bar{x}')}{P(s=1 \mid \bar{x}, \bar{x}')} \quad \ulcorner d.sep(3) \lrcorner$$

$$= \frac{\sigma(\bar{w} \cdot \bar{x}) \sigma(\bar{w} \cdot \bar{x}')}{\sum_{y,y'} P(s=1, y, y' \mid \bar{x}, \bar{x}')}$$

$$= \frac{\sigma(\bar{w} \cdot \bar{x}) \sigma(\bar{w} \cdot \bar{x}')}{\sum_{y,y'} \underbrace{P(s=1 \mid y, y', \bar{x}, \bar{x}')}_{(=1 \text{ iff } y=y') \Rightarrow y=y'=0 \text{ \& } y=y'=1} P(y \mid \bar{x}) P(y' \mid \bar{x}')} \rightarrow \text{same d.sep(3) as last step}$$

$$= \frac{\sigma(\bar{w} \cdot \bar{x}) \sigma(\bar{w} \cdot \bar{x}')}{P(y=0 \mid \bar{x}) P(y'=0 \mid \bar{x}) + P(y=1 \mid \bar{x}) P(y'=1 \mid \bar{x})}$$

$$= \frac{\sigma(\bar{w} \cdot \bar{x}) \sigma(\bar{w} \cdot \bar{x})'}{(1-\sigma(\bar{w} \cdot \bar{x}))(1-\sigma(\bar{w} \cdot \bar{x}')) + \sigma(\bar{w} \cdot \bar{x}) \sigma(\bar{w} \cdot \bar{x}')}$$

(b) Similarly,

$P(y=1, y'=0 \mid \bar{x}, \bar{x}', s=0)$

$$= \frac{P(y=1 \mid \bar{x}) P(y'=0 \mid \bar{x}')}{\sum_{y,y'} P(s=0, y, y' \mid \bar{x}, \bar{x}')}$$

$\hookrightarrow$ need $y=0, y'=1$ & $y=1, y'=0$

$$= \frac{\sigma(\bar{w} \cdot \bar{x})(1-\sigma(\bar{w} \cdot \bar{x}'))}{P(y=0 \mid \bar{x}) P(y'=1 \mid \bar{x}) + P(y=1 \mid \bar{x}) P(y'=0 \mid \bar{x})}$$

$$= \frac{\sigma(\bar{w} \cdot \bar{x})(1-\sigma(\bar{w} \cdot \bar{x}'))}{(1-\sigma(\bar{w} \cdot \bar{x}))\sigma(\bar{w} \cdot \bar{x}') + \sigma(\bar{w} \cdot \bar{x})(1-\sigma(\bar{w} \cdot \bar{x}'))}$$

(c) $\underbrace{BB}_{s=1 \Rightarrow y=y'}\ \underbrace{AC}_{s=0 \Rightarrow y \neq y'}$

(d) $\mathcal{L}(\bar{w}) = \sum_t \log P(s_t \mid \bar{x}_t, \bar{x}_t')$

$$= \sum_t s_t \log P(s_t=1 \mid \bar{x}_t, \bar{x}_t') + (1-s_t) \log P(s_t=0 \mid \bar{x}_t, \bar{x}_t')$$

$$= \sum_t s_t \log\left[(1-\sigma(\bar{w} \cdot \bar{x}_t))(1-\sigma(\bar{w} \cdot \bar{x}_t')) + \sigma(\bar{w} \cdot \bar{x}_t)\sigma(\bar{w} \cdot \bar{x}_t')\right] \quad \ulcorner \text{from denom. in (a), (b)} \lrcorner$$

$$+ (1-s_t) \log\left[(1-\sigma(\bar{w} \cdot \bar{x}_t))\sigma(\bar{w} \cdot \bar{x}_t') + \sigma(\bar{w} \cdot \bar{x}_t)(1-\sigma(\bar{w} \cdot \bar{x}_t'))\right]$$

(e) $\frac{\partial \mathcal{L}}{\partial \vec{w}} = \sum_t \Big[\bar{y}_t \frac{\sigma(\bar{w}\cdot\bar{x}_t)}{\sigma(\bar{w}\cdot\bar{x}_t)} \sigma(-\bar{w}\cdot\bar{x}_t)\bar{x}_t + (1-\bar{y}_t) \frac{\sigma(-\bar{w}\cdot\bar{x}_t)}{\sigma(-\bar{w}\cdot\bar{x}_t)} \sigma(\bar{w}\cdot\vec{x}_t)\bar{x}_t$

$+ \bar{y}_t' \frac{\sigma(\bar{w}\cdot\bar{x}_t')}{\sigma(\bar{w}\cdot\bar{x}_t')} \sigma(-\bar{w}\cdot\bar{x}_t')\bar{x}_t' + (1-\bar{y}_t') \frac{\sigma(-\bar{w}\cdot\bar{x}_t')}{\sigma(-\bar{w}\cdot\bar{x}_t')} \sigma(\bar{w}\cdot\bar{x}_t')\bar{x}_t' \Big]$

$= \sum_t \left[(\bar{y}_t - \sigma(\bar{w}\cdot\bar{x}_t))\bar{x}_t + (\bar{y}_t' - \sigma(\bar{w}\cdot\bar{x}_t'))\bar{x}_t' \right]$

and the expression in [] goes into the bracket.

5. (a) $\alpha_{it} = P(Y_t = i \mid Y_0, \bar{x}_1, \cdots, \bar{x}_t)$

$= \sum_{j\in\{0,1\}} P(Y_t = i, Y_{t-1} = j \mid Y_0, \bar{x}_1, \cdots, \bar{x}_t)$

$= \sum_j P(Y_{t-1} = j \mid Y_0, \bar{x}_1, \cdots, \bar{x}_t) P(Y_t = i \mid Y_0, \bar{x}_1, \cdots, \bar{x}_t, Y_{t-1} = j)$

$= \sum_j \underset{d\text{-sep}(3)}{P(Y_{t-1} = j \mid Y_0, \bar{x}_1, \cdots, \bar{x}_{t-1})} \underset{d\text{-sep}(1)}{P(Y_t = i \mid \bar{x}_t, Y_{t-1} = j)}$

$= \begin{cases} \alpha_{0(t-1)}\sigma(\bar{w}_0\cdot\bar{x}_t) + \alpha_{1(t-1)}\sigma(\bar{w}_1\cdot\bar{x}_t), & \text{if } i=1 \\ \alpha_{0(t-1)}(1-\sigma(\bar{w}_0\cdot\bar{x}_t)) + \alpha_{1(t-1)}(1-\sigma(\bar{w}_1\cdot\bar{x}_t)), & \text{if } i=0. \end{cases}$

(b) $\ell^*_{it} = \max_{y_1,\cdots,y_{t-1}} \left[\log P(y_1, \cdots, y_t = i \mid y_0, \bar{x}_1, \cdots, \bar{x}_t) \right]$

$= \max_{y_1,\cdots,y_{t-1}} \left[\log P(y_1, \cdots, y_{t-1} \mid y_0, \bar{x}_1, \cdots, \bar{x}_t) + \log P(y_t = i \mid y_0, \bar{x}_1, \cdots, \bar{x}_t, y_1, \cdots, y_{t-1}) \right]$ PR

$= \max_{y_1,\cdots,y_t} \underset{d\text{-sep}(3)}{\log P(y_1, \cdots, y_{t-1} \mid y_0, \bar{x}_1, \cdots, \bar{x}_{t-1})} + \max_{y_1,\cdots,y_t} \underset{d\text{-sep}(1)}{\log P(y_t = i \mid \bar{x}_t, y_{t-1})}$

$= \ell^*_{i(t-1)} + \max(\log\sigma(\bar{w}_0, \bar{x}_t), \log\sigma(\bar{w}_1, \bar{x}_t), \log(1-\sigma(\bar{w}_0, \bar{x}_t)), \log(1-\sigma(\bar{w}_1, \bar{x}_t)))$.

(c) $\alpha_{it} = \sum_{i\in\{0,1\}} P(Y_t = i, Y_{t+1} = j \mid Y_0, \bar{x}_1, \cdots, \bar{x}_t)$

$= \sum_{i\in\{0,1\}} \underbrace{P(Y_t = i \mid Y_0, \bar{x}_1, \cdots, \bar{x}_{t+1}) P(Y_{t+1} = j \mid Y_t = i, \bar{x}_{t+1})}_{(*)}$

Take log we have $(*) = \ell^*_{it} + \log P(Y_{t+1} = j \mid Y_t = i, \bar{x}_{t+1})$.

this is analogous to the viterbi algo. for HMM,

so $\Phi_{t+1}(j) = \operatorname{argmax}_{i\in\{0,1\}} \left[\ell^*_{it} + \log P(Y_{t+1} = j \mid Y_t = i, \bar{x}_{t+1}) \right]$

and $y_t^* = \Phi_{t+1}(y_{t+1}^*)$.

HW8 Code

November 23, 2021

[1]:
```
import numpy as np

pid = []

with open("hw8_ids.txt", "r") as ids:
    for line in ids.readlines():
        pid.append(line[:-1])

movie = []
with open("hw8_movies.txt", "r") as mvs:
    for line in mvs.readlines():
        movie.append(line[:-1])

rating = []
with open("hw8_ratings.txt", "r") as rts:
    for line in rts.readlines():
        line = line.strip('\n')
        line = line.split(' ')
        rating.append(line)

rating = np.array(rating)
print(rating.shape)

index = {}
for i in range(rating.shape[1]):
    saw = 0
    rec = 0
    for j in range(rating.shape[0]):
        if rating[j][i] != '?':
            saw += 1
        if rating[j][i] == '1':
            rec += 1
    index[i] = rec / saw

index_sorted = sorted(index.items(), key=lambda x: x[1])
movie_sorted = []
for i in range(len(index_sorted)):
    movie_sorted.append(movie[index_sorted[i][0]])
```

```python
print(movie_sorted)

# 8.1.e
probR = np.loadtxt("hw8_probR_init.txt")
probZ = np.loadtxt("hw8_probZ_init.txt").flatten()


# returns Omega_t
def seen(t) -> list:
    res = []
    for rate in rating[t]:
        if rate != '?':
            res.append(1)
        else:
            res.append(0)
    return res


# Z R are ndarray
def prior(Z, R, t):
    K = Z.shape[0]  # 4
    O_t = seen(t)
    J = len(movie)  # 76
    sum = 0

    for i in range(K):
        prod = 1
        for j in range(J):
            if O_t[j] == 0:
                continue
            r_jt = int(rating[t][j])
            if r_jt == 1:
                prod *= R[j][i]
            else:
                prod *= 1 - R[j][i]
        sum += Z[i] * prod
    return sum


def rou_it(Z, R, i, t):
    denom = prior(Z, R, t)
    prod = 1
    O_t = seen(t)
    J = len(movie)
    for j in range(J):
        if O_t[j] == 0:
            continue
```

```
        r_jt = int(rating[t][j])
        if r_jt == 1:
            prod *= R[j][i]
        else:
            prod *= 1 - R[j][i]
    num = Z[i] * prod
    return num / denom


def likelihood(Z, R):
    sum = 0
    T = len(pid)
    for t in range(T):
        sum += np.log(prior(Z, R, t))
    return sum / T


def update(Z, R, iter):
    res_Z = np.copy(Z)
    res_R = np.copy(R)
    K = Z.shape[0]  # 4

    T = len(pid)  # 362
    J = len(movie)  # 76

    for k in range(iter):  # 256
        if k in [0, 1, 2, 4, 8, 16, 32, 64, 128, 256]:
            print(k)
            ll = likelihood(res_Z, res_R)
            print(ll)

        # rou_it = rou_it_mat[t][i]
        rou_it_mat = np.zeros((362, 4))
        for t in range(362):
            for i in range(4):
                rou_it_mat[t][i] = rou_it(res_Z, res_R, i, t)

        # update P(Z=i)

        for i in range(K):  # 4
            sum = 0
            for t in range(T):
                sum += rou_it_mat[t][i]
            res_Z[i] = sum / T
        # Update P(R_j=1|Z=i)
        for j in range(J):  # 76
            for i in range(K):  # 4
```

```python
                denom = 0
                for t in range(T):
                    denom += rou_it_mat[t][i]
                num = 0
                for t in range(T):  # 362
                    if seen(t)[j] == 1:
                        if int(rating[t][j]) == 1:
                            num += rou_it_mat[t][i]
                        else:
                            continue
                    else:
                        num += rou_it_mat[t][i] * res_R[j][i]
                res_R[j][i] = num / denom
    return res_Z, res_R


# takes about 10 minutes to run :(
final_Z, final_R = update(probZ, probR, 257)

prob = [0] * len(movie)
pid = np.array(pid)

row = int(np.where(pid == 'A15058075')[0])
# row = 344
rate = rating[row]
binary_seen = seen(row)
for k in range(len(prob)):
    if binary_seen[k] == 1:
        continue
    sum = 0
    for i in range(4):
        sum += rou_it(final_Z, final_R, i, row) * final_R[k][i]
    prob[k] = sum
index_mine = {}
for i in range(len(prob)):
    if prob[i] == 0:
        continue
    index_mine[i] = prob[i]

index_mine_sorted = sorted(index_mine.items(), key=lambda x: x[1], reverse=True)
movie_mine_sorted = []
for i in range(len(index_mine_sorted)):
    movie_mine_sorted.append(movie[index_mine_sorted[i][0]])
print(movie_mine_sorted)
```

```
(362, 76)
['I_Feel_Pretty', 'Fifty_Shades_of_Grey', 'Hustlers', 'The_Last_Airbender',
'Magic_Mike', 'Fast_&_Furious:_Hobbs_&_Shaw', 'The_Shape_of_Water',
```

```
'Prometheus', 'Phantom_Thread', 'World_War_Z', 'Star_Wars:_The_Force_Awakens',
'Rocketman', 'Chappaquidick', 'Bridemaids', 'Man_of_Steel', 'American_Hustle',
'Terminator:_Dark_Fat', 'Room', 'Good_Boys', 'Pokemon_Detective_Pikachu',
'Fast_Five', 'Mad_Max:_Fury_Road', 'Drive', 'Us', 'The_Help', 'Pitch_Perfect',
'Jurassic_World', 'Frozen', 'X-Men:_First_Class', 'The_Revenant', 'Ex_Machina',
'Avengers:_Age_of_Ultron', 'La_La_Land', 'Midnight_in_Paris',
'Manchester_by_the_Sea', 'Once_Upon_a_Time_in_Hollywood',
'Three_Billboards_Outside_Ebbing', 'Darkest_Hour', 'The_Great_Gatsby',
'Dunkirk', 'Her', 'Captain_America:_The_First_Avenger',
'The_Girls_with_the_Dragon_Tattoo', 'Ready_Player_One', 'Hidden_Figures',
'The_Hateful_Eight', 'Thor', 'Toy_Story_3', 'The_Hunger_Games',
'12_Years_a_Slave', 'Iron_Man_2', 'The_Perks_of_Being_a_Wallflower', 'Joker',
'Les_Miserables', '21_Jump_Street', 'Spiderman:_Far_From_Home', 'Black_Swan',
'Parasite', 'The_Avengers', 'The_Farewell', 'Django_Unchained',
'Now_You_See_Me', 'Avengers:_Endgame', 'Avengers:_Infinity_War',
'Wolf_of_Wall_Street', 'The_Lion_King', 'Gone_Girl',
'Harry_Potter_and_the_Deathly_Hallows:_Part_1', 'The_Social_Network',
'Harry_Potter_and_the_Deathly_Hallows:_Part_2', 'The_Theory_of_Everything',
'Interstellar', 'The_Martian', 'The_Dark_Knight_Rises', 'Shutter_Island',
'Inception']
0
-27.03581500351123
1
-17.5604038243144
2
-16.002362630627825
4
-15.060597317892249
8
-14.501649272824999
16
-14.26378857143746
32
-14.180178075094306
64
-14.170077781591038
128
-14.163960358152185
256
-14.163692439007862
['Shutter_Island', 'The_Social_Network', 'Avengers:_Age_of_Ultron',
'The_Theory_of_Everything', 'The_Farewell', 'Now_You_See_Me', 'The_Martian',
'21_Jump_Street', 'The_Perks_of_Being_a_Wallflower', 'Django_Unchained',
'Midnight_in_Paris', 'The_Hateful_Eight', 'Toy_Story_3', 'Parasite',
'Black_Swan', 'The_Last_Airbender', 'X-Men:_First_Class', 'Us',
'Ready_Player_One', 'The_Girls_with_the_Dragon_Tattoo', 'Pitch_Perfect',
'Frozen', 'Man_of_Steel', 'Hidden_Figures', 'Jurassic_World', 'Fast_Five',
'Rocketman', 'Ex_Machina', 'Chappaquidick', 'American_Hustle', 'The_Help',
```

```
'Good_Boys', 'Magic_Mike', 'Darkest_Hour', 'Three_Billboards_Outside_Ebbing',
'The_Revenant', 'Mad_Max:_Fury_Road', 'The_Shape_of_Water', 'Her',
'Fifty_Shades_of_Grey', 'Drive', 'Bridemaids', 'Phantom_Thread', 'Hustlers',
'Prometheus', 'I_Feel_Pretty']
```

[ ]:

```

```