Assignment 4 B20001 Aditya Prakash

1. (MLE)

1. To maximize prob. of 2-A, 10-B, 60-C, 40-F with corresponding frequencies.

$P = \theta^2 \times (3\theta)^{10} \times \left(\frac{1}{2}\right)^{60} \times \left(\frac{1}{2} - 4\theta\right)^{40}$

$L = 2\ln\theta + 10\ln 3\theta + 40\ln\left(\frac{1}{2} - 4\theta\right) + 60\ln\frac{1}{2}$

$\frac{\partial L}{\partial \theta} = 0$ ; for max.

$\frac{2}{\theta} + \frac{10}{\theta} + \frac{40}{\frac{1}{2} - 4\theta} \times -4 = 0$

$\frac{12}{\theta} = \frac{4 \times 40}{\frac{1}{2} - 4\theta} \Rightarrow \frac{3}{2} - 12\theta = 40\theta \Rightarrow \theta = \frac{3}{104}$

2. Histogram distribution

$\lambda e^{-\lambda x}$

$P = (\lambda e^{-\lambda})^1 (\lambda e^{-\lambda \cdot})^1 (\lambda e^{-\lambda \times 3})^2 (\lambda e^{-\lambda \times 4})$

$L = \ln\lambda - \lambda + \ln\lambda - 2\lambda + 2\ln\lambda - 6\lambda + \ln\lambda - 4\lambda = 5\ln\lambda - 13\lambda$

$\frac{\partial L}{\partial \lambda} = 0 \Rightarrow \frac{5}{\lambda} - 13 = 0 \Rightarrow \lambda = \frac{5}{13}$

3. $P = (\theta^2)^{k_1} (2\theta(1-\theta))^{k_2} ((1-\theta)^2)^{k_3}$

$l = 2k_1\ln\theta + k_2\ln 2\theta + k_2\ln(1-\theta) + 2k_3\ln(1-\theta)$

$\frac{\partial L}{\partial \theta} = \frac{2k_1}{\theta} + \frac{k_2}{\theta} - \frac{k_2}{1-\theta} - \frac{2k_2}{1-\theta} = 0 + \frac{2k_3}{1-\theta}$

$(2k_1 + k_2) - 2k_1\theta - 2k_2\theta = k_2\theta + 2k_2\theta + 2k_3\theta$

$\Rightarrow \theta = \frac{2k_1 + k_2}{2k_1 + 5k_2 + 2k_3}$

2. (GMM)

2011 $K = 1, 2 \ldots K$ player plays $M_t$ and wins $W_t$ at day $t$

$P = {}^{M_t}C_{W_t}\, P_k^{W_t} (1-P_k)^{M_t - W_t}$

$P_k$ = for each player probability that she wins independent of day.

$C_t \sim \pi$ which player plays on $t$ out of $N$

$$\gamma_k(n) = P(k|n) = \frac{P(k)\,P(n|k)}{P(n)}$$

1st way.

$$= \frac{\pi_k N(n \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j N(n \mid \mu_j, \Sigma_j)} \qquad \pi_k = \frac{M_k}{N}$$

$\therefore$

$$\theta_t^{(i)}[k] = P(C_t = k \mid W = W_t) = \frac{P(C_t = t)\, P(W = W_t \mid C_t = t)}{\sum_{k=1}^{K} P(C_t = t_k)\, P(W = W_{t_k} \mid C_t = t_k)}$$

$$= \frac{\pi[k]\, {}^{M_t}C_{W_t} (P_k)^{W_t} (1-P_k)^{M_t - W_t}}{\sum_{k=1}^{K} \pi[k]\, {}^{M_t}C_{W_t} P_k^{M_t} (1-P)^{M_t - W_t}}$$

1) E step:

$$a_t^{(i)}[k] = \frac{\lambda^{(i-1)}[k]\, B(W_t, M_t, P_k^{(i-1)})}{\sum_{k=1}^{K} \lambda^{(t-1)}[k]\, B(W_t, M_t, P_k^{(i-1)})}$$

DATE
PAGE

2) log likelihood $= \prod_{t=1}^{n} P(N = N_t)$

$$= \prod_{t=1}^{n} \sum_{k=1}^{K} \pi_k \, {}^{M_t}C_{w_t} P_k^{N_t} (1 - P_k)^{M_t - W_t)}$$

$$ll(\pi_k, P_k) = \sum_{t=1}^{n} \ln\left( \sum_{k=1}^{K} \pi_k \, {}^{M_L}C_{W_t} P_k^{W_t} (1-P_k)^{M_t - N_t} \right)$$

$$\frac{\partial ll}{\partial P_k} = 0 \Rightarrow \sum_{t=1}^{n} \frac{\pi_k \, {}^{M_L}C_{W_t} N_t P_k^{(N_t+1)} (1-P_k)^{M_t - W_t - 1} \, 1 - (M_t - W))}{\Delta} = 0 .$$

$$\Rightarrow P_k \sum_{t=1}^{N} Q_t^{(i)} [k] N_t - \sum_{t=1}^{N} Q_t^{(i)} [k] N_t)$$

$$= P_k \sum_{t=1}^{N} Q_i^{(i)} [k] (M_t \; n_t) .$$

$$P_k = \frac{\sum_{t=1}^{N} Q_t^{(i)} [k] N_t}{\sum_{t=1}^{N} Q^{(i)} [k] M_t}$$

and. $\pi_k = \frac{N_k}{N}$

$$\frac{\partial ll}{\partial \pi_k} = \sum_{t=1}^{n} \frac{1}{D} \, {}^{M_t}C_{W_t} P_k^{N_t} (1-P_k)^{M_t - N_t} = 0 .$$

Q. - 2

(a) $\gamma_{nk}^{t} = P(L_n = k \mid x_n, y_n, \theta^{(t-1)})$

$(x_i, y_i) - x_i \to \mathbb{R}^d$

$a_j^T x_i = y_i \quad a_1, a_2 \cdots a_m$

$a_i \to \mathbb{R}^d$.

$$P(\vec{x}, y \mid \theta) = \sum_{j=1}^{m} \pi_j \frac{1}{\sqrt{2\pi\alpha^2}} e^{\left( \frac{-(a_j^T x - y)^2}{-2\alpha^2} \right)}.$$

$\theta = (\pi_{1:m}, \vec{a}_{1:m}) \quad \sum \pi_j = 1. \quad \alpha > 0.$

$$\gamma_{nk}^{t} = \frac{P(t_n = k) P(x_n, y_n \theta^{(t-1)} \mid t_n = k)}{P(x_n, y_n, \theta^{(t-1)})}$$

$$= \frac{\pi_k^{(t-1)} \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(\alpha_k^{(t-1)T} x - y)^2}{2\alpha^2}}}{\sum_{k=1}^{m} \pi_k^{(t-1)} \frac{1}{\sqrt{2\pi a^2}} e^{\frac{-(a_k^{(t-1)T} x - y)^2}{2\alpha^2}}}$$

(b) obtain optimal $\pi_j^{(t)}$ & $a_j^{(t)}$

$$\frac{\partial LL}{\partial \pi_j} = 0, \quad + \quad \frac{\partial LL}{\partial a_j} = 0$$

$$LL = \sum_{n=1}^{N} \ln \left( \sum_{j=1}^{N} \pi_j \frac{1}{\sqrt{2\pi a^2}} e^{-\frac{(a_j^T x_n - y_n)^2}{2\alpha^2}} \right)$$

DATE
PAGE

$$\frac{\partial LL}{\partial \pi_i} = \sum_{n=1}^{N} \frac{\frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(a_j^T x_n - y_n)^2}{2\alpha^2}}}{\sum_{j=1}^{M} \pi_j \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(a_j^T x_n - y_n)^2}{2\alpha^2}}} = 0$$

$$\Rightarrow \sum_{n=1}^{N} \frac{1}{\pi_j} \gamma_{nj} - 1 = 0 .$$

$$\pi_j = \frac{\sum_{n=1}^{N} \gamma_{nj}}{N} \Rightarrow \pi_j^t = \frac{\sum_{n=1}^{N} \gamma_{nj}^{(t)}}{N}$$

$$\frac{\partial LL}{\partial a_j} = 0 \Rightarrow \sum_{n=1}^{N} \frac{1}{D} \pi_j \frac{1}{\sqrt{2\pi\alpha^2}} e^{-\frac{(a_j^T x_n - y_n)^2}{2\alpha^2}} \times \frac{-1}{\alpha^2} \times (a_j^T x_n - y_n) x_n = 0 .$$

$$\Rightarrow \sum_{n=1}^{N} \gamma_{nj} y_n x_n = \sum_{n=1}^{N} \gamma_{nj} a_j^T x_n x_n .$$

$$(a_j^T x_n) x_n = x_n \cdot x_n^T \cdot a_j$$

$$a_j = \left( \sum_{n=1}^{N} \gamma_{nj} x_n x_n^T \right)^{-1} \left( \sum_{n=1}^{N} (\gamma_{nj} y_n) x_n \right) .$$

$$a_j^t = \left( \sum_{n=1}^{N} \gamma_{nj}^{(t)} x_n x_n^T \right)^{-1} \left( \sum_{n=1}^{N} \gamma_{nj}^{(t)} y_n x_n \right)$$

## Complexity of SVM

general SVM Alg – iteration $m$ times :
iterate over all dp . $n$ :
update $w, b$ .

Overall complexity $= O(n \times m)$
given it converges after m iters.

in terms of Learn. rate or var. $\eta$ learnrate:
complexity. $O(LR \times \partial(LR) \cdot n \times m)$.

- comparision of classifiers.

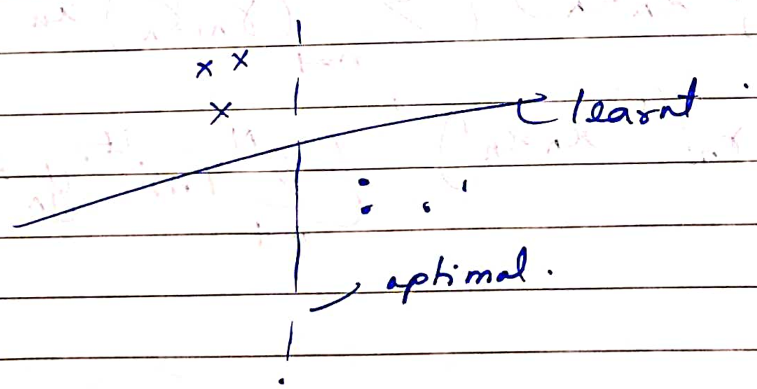
→ Logistic Regres.
- linear boundary.

Obj: $\sum y_i \log\left(\frac{P x_i}{1-P(x_i)}\right) + \log(1-P(x_i))$

learn. soft margin
learn hyperplane. (need not be optimal.

→ Linear LS.C
objective. $\arg\min_{w,b} \sum (y - \sum x_i w + b)^2$

→ learns the horizontal boundary if there's vertical & horizontal bary parallel.



Bayes:

$$\arg\max \prod_{k=1}^{d} P(x_k | C_i)$$

Bayes ds. clas soft clustr:
for given test dat, calculat P(test data in a cluster)

SVM
obj: $\text{Argmin}\left[ \lambda \|w\|^2 + \frac{1}{n} \sum_{i=1}^{n} \max(0, 1 - y_i(wx_i - b)) \right]$

SVM can learn soft or hard margin
SVM learns optimal hyperplane, i.e. maximum distance from all data pts with spread the
clusters as much as possible.