Part 1

Question 1

2019年03月23日 CSC411 HW6

1. $$\ell = \sum_{i=1}^{N}\sum_{k=1}^{K} r_k^{(i)}\left[\log \Pr(z^i=k) + \log P(x^i \mid z^i=k)\right] + \log P(\pi) + \log P(\theta)$$

$z \sim \text{Multinomial}(\pi)$

$x_j \mid z=k \sim \text{Bernoulli}(\theta_{k,j})$

$\theta_{k,j} \sim \text{Beta}(a,b)$ e.g. $P(\theta_{k,j}) \propto \theta_{k,j}^{a-1}(1-\theta_{k,j})^{b-1}$ prior over θ

$\pi \sim \text{Dirichlet}(c,\dots,c)$ e.g. $P(\pi) = \pi_1^{(c-1)} \cdots \pi_K^{(c-1)}$ prior over class

How to derive ℓ above?

$$\log P(\theta,\pi \mid D) = \log P(D\mid\theta)P(\theta) = \log P(D\mid\theta) + \log P(\theta) + \log P(\pi)$$

$$P(D\mid\theta) = P(x^1,y^1) \cdots P(x^N,y^N)$$ assuming we knew labels

$$= P(x^1\mid y^1)P(y^1) \cdots P(x^N\mid y^N)P(y^N)$$

$$\log P(\theta\mid D) = \sum_{i=1}^{N} \log P(x^i\mid y^i) + \sum_{i=1}^{N}\log P(y^i) + \log P(\theta) + \log P(\pi)$$

$$= \sum_{k=1}^{K}\sum_{i=1}^{N} \delta(y^i=k)\left[\log P(x^i\mid y^i=k) + \log P(y^i=k)\right] + \log(P(\theta)) + \log P(\pi)$$

Plug in $r_k^i = P(y^i=k\mid x^i)$ for $\delta(y^i=k)$

$$r_k^i = P(y^i=k\mid x^i) = \frac{P(x^i\mid y^i=k)P(y^i=k)}{P(x^i)} = \frac{\prod_{j=1}^{D}\theta_{k,j}^{x_j^i}(1-\theta_{k,j})^{1-x_j^i}\cdot \pi_k}{\sum_{p=1}^{K}\prod_{j=1}^{D}\theta_{p,j}^{x_j^i}(1-\theta_{p,j})^{1-x_j^i}\pi_p}$$

r_k^i is the "responsibility" the k^{th} cluster has to the i^{th} data point

$$\ell = \sum_{i=1}^{N}\sum_{k=1}^{K} \underbrace{\frac{\prod_{j=1}^{D}\theta_{k,j}^{x_j^i}(1-\theta_{k,j})^{1-x_j^i}\cdot\pi_k}{\sum_{p=1}^{K}\prod_{j=1}^{D}\theta_{p,j}^{x_j^i}(1-\theta_{p,j})^{1-x_j^i}\pi_p}}_{r_k^i} \left[\log\pi_k + \sum_{j=1}^{D} x_j^i\log\theta_{k,j} + (1-x_j^i)\log(1-\theta_{k,j})\right] + \sum_{k=1}^{K} c^{-1}\log\pi_k$$

θ,π here are constants

$$+ \sum_{k=1}^{K}\sum_{j=1}^{D} (a-1)\log\theta_{k,j} + (b-1)\log(1-\theta_{k,j})$$ assuming indep $\theta_{k,j}$

Fact: $\sum_{k=1}^{K} r_k^i = 1$

Maximize ℓ subject to $\sum_{k=1}^{K}\pi_k = 1$ $\therefore$ $\sum_{k=1}^{K}\pi_k - 1 = 0 \equiv g$

$$\frac{d\ell}{d\pi_b} = \sum_{i=1}^{N}\frac{r_b^i}{\pi_b} + \frac{c}{\pi_b} = \lambda 1 = \lambda\frac{dg}{d\pi_b}$$

$$\frac{1}{\pi_b}\left(\sum_{i=1}^{N} r_b^i + c^{-1}\right) = \lambda$$

$$\pi_b = \frac{c^{-1} + \sum_{i=1}^{N} r_b^i}{\lambda}$$

$$\sum_{k=1}^{K}\frac{c^{-1}+\sum_{i=1}^{N} r_k^i}{\lambda} = 1 \Rightarrow \lambda = \sum_{k=1}^{K}\left(c^{-1}+\sum_{i=1}^{N} r_k^i\right) \cdot 1$$

$$\lambda = Kc^{-1} + \sum_{k=1}^{K}\sum_{i=1}^{N} r_k^i = Kc^{-1} + \sum_{i=1}^{N}\sum_{k=1}^{K} r_k^i = Kc^{-1} + \sum_{i=1}^{N} 1 = Kc + N$$

$$\boxed{\therefore \pi_b = \frac{(c-1) + \sum_{i=1}^{N} r_b^i}{K(c-1) + N}}$$

Note: as expected $\sum_{k=1}^{K}\pi_k = 1$

$$\frac{dl}{d\theta_{p,q}} = \sum_{i=1}^{N} r_p^i \left(\frac{x_q^i}{\theta_{pq}} + \frac{x_q^i - 1}{1-\theta_{pq}} \right) + \frac{a-1}{\theta_{p,q}} - \frac{b-1}{1-\theta_{p,q}}$$

$$0 = \frac{a-1+\sum_{i=1}^{N} r_p^i x_q^i}{\theta_{p,q}} + \frac{1-b+\sum_{i=1}^{N} r_p^i (x_q^i - 1)}{1-\theta_{p,q}}$$

$$0 = (1-\theta_{p,q})(a-1+f) + \theta_{p,q}(1-b+g), \quad f = \sum_{i=1}^{N} r_p^i x_q^i \quad g = \sum_{i=1}^{N} r_p^i (x_q^i - 1)$$

$$0 = (a+f-1) + (1-f-a)\theta_{p,q} + (+1+g-b)\theta_{p,q}$$

$$0 = (a+f-1) + (1-f-a+b+g+1)\theta_{p,q}$$

$$\frac{1-a-f}{-b+g-f-a+2} = \theta_{p,q}$$

$$g-f = \sum_{i=1}^{N} r_p^i (x_q^i - 1) - r_p^i x_q^i$$
$$= -\sum_{i=1}^{N} r_p^i$$

$$\theta_{p,q} = \frac{1-a-\sum_{i=1}^{N} r_p^i x_q^i}{-(b+a) - \sum_{i=1}^{N} r_p^i + 2}$$

$$\boxed{\theta_{p,q} = \frac{\sum_{i=1}^{N} (r_p^i x_q^i) + (a-1)}{\left(\sum_{i=1}^{N} r_p^i\right) + (a+b) - 2}}$$

class

feature

How to vectorize?

$$\underset{N\times K}{R} = \begin{bmatrix} - r^1 - \\ \vdots \\ - r^N - \end{bmatrix} \qquad \underset{N\times D}{X} = \begin{bmatrix} - x^1 - \\ \vdots \\ - x^N - \end{bmatrix}$$

Say $R = \begin{bmatrix} {}^1r & \cdots & {}^Kr \\ | & & | \end{bmatrix}$ — each col. is responsibilities of K^{th} class of all training examples

$X = \begin{bmatrix} {}^1x & \cdots & {}^Dx \\ | & & | \end{bmatrix}$ — same idea.

$$R^T X = \begin{bmatrix} - {}^1r - \\ \vdots \\ - {}^Kr - \end{bmatrix} \begin{bmatrix} {}^1x & \cdots & {}^Dx \\ | & & | \end{bmatrix} = \begin{bmatrix} {}^1r\,{}^1x & \cdots & {}^1r\,{}^Dx \\ \vdots & & \\ {}^Kr\,{}^1x & \cdots & {}^Kr\,{}^Dx \end{bmatrix}$$

${}^pr\,{}^qx$

Then divide each element of a row by $\sum_{i=1}^{N} r_p^i$.

Question 2

Part 1 values:

('pi[0]', 0.084999999999999923)

('pi[1]', 0.12999999999999987)

('theta[0, 239]', 0.64271062271062329)

('theta[3, 298]', 0.46573612495845823)

Part 2

Question 1

Part 2

1. $p(z=k|x) = \dfrac{P(x|z=k)P(z=k)}{P(x)} = \dfrac{P(x|z=k)P(z=k)}{\sum_{p=1}^{K} P(x|z=p)P(z=p)}$

$$\boxed{p(z=k|x) = \frac{\prod_{j=1}^{D} \theta_{k,j}^{x_j}(1-\theta_{k,j})^{1-x_j}\cdot \pi_k}{\sum_{p=1}^{K}\left[\prod_{j=1}^{D} \theta_{p,j}^{x_j}(1-\theta_{p,j})^{1-x_j}\right]\pi_p}}$$

↳ assumes fully observed though.

$$Pr(z=k|X_{obs}) = \sum_{X_{HIDDEN}} Pr(z=k, X_{HIDDEN}|X_{OBS})$$

$$= \sum_{X_{HIDDEN}} Pr(z=k|X_{HIDDEN}, X_{OBS})\, Pr(X_{HIDDEN}|X_{OBS})$$

By independance.

$$= \sum_{X_{HIDDEN}} Pr(z=k|X_{HIDDEN}, X_{OBS})\, Pr(X_{HIDDEN})$$

$$= \sum_{X_{HIDDEN}} \frac{Pr(X_{HIDDEN}, X_{OBS}|z=k)Pr(z=k)}{Pr(X_{HIDDEN}, X_{OBS})} \cdot Pr(X_{HIDDEN})$$

independence

$$= \sum_{X_{HIDDEN}} \frac{Pr(X_{HIDDEN}|z=k)Pr(X_{OBS}|z=k)Pr(z=k)}{\cancel{Pr(X_{HIDDEN})}\,Pr(X_{OBS})} \cancel{Pr(X_{HIDDEN})}$$

$$= \frac{Pr(X_{OBS}|z=k)Pr(z=k)}{Pr(X_{OBS})} \overbrace{\sum_{X_{HIDDEN}} Pr(X_{HIDDEN}|z=k)}^{1}$$

$$\boxed{Pr(z=k|X_{OBS}) = \frac{Pr(X_{OBS}|z=k)\cdot Pr(z=k)}{Pr(X_{OBS})}}$$

— e.g. we only multiply $\theta_{k,j}$ for x_j observed.

Question 2

Vectorization:

$$Pr(x^1|z=k) = \prod_{j=1}^{D} \theta_{k,j}^{x_j^1} (1-\theta_{k,j})^{1-x_j^1}$$

log ↓

$$= \sum_{j=1}^{D} x_j^1 \theta_{k,j} + \sum_{j=1}^{D} (1-x_j^1)(1-\theta_{k,j})$$

$$X = \begin{bmatrix} — x^1 — \\ \vdots \\ — x^N — \end{bmatrix} \qquad \Theta = \begin{bmatrix} — \theta_1 — \\ \vdots \\ — \theta_K — \end{bmatrix}$$

$Pr(x^1 | z = j), j \in (1, K)$

$$X\Theta^T = \begin{bmatrix} — x^1 — \\ \vdots \\ — x^N — \end{bmatrix} \begin{bmatrix} | & & | \\ \theta_1 & \cdots & \theta_K \\ | & & | \end{bmatrix} = \begin{bmatrix} x^1\theta_1 & \cdots & x^1\theta_K \\ \vdots & & \\ x^N\theta_1 & \cdots & x^N\theta_K \end{bmatrix}$$

Posterior Predictive Mean

$$p(x|X) = \int_\theta p(x|\theta)p(\theta|X)d\theta$$

How to do this?

$$E[p(x|X)] =$$

$$Pr(x_j^{(i)}|z=K) = \theta_{k,j}^{x_j^i}(1-\theta_{k,j})^{1-x_j^i}$$

$$E[x_j^{(i)}] = (1)\theta_{k,j} + (0)(1-\theta_{k,j}) = \theta_{k,j}$$

→ take the argmax of posterior to get K.

Question 3

Part 2 values:
('R[0, 2]', 0.1748895149211743)
('R[1, 0]', 0.6885376761092308)
('P[0, 183]', 0.744545579523189)
('P[2, 628]', 0.4591808320583727)

Part 3

Output of average log probability by digit class for model trained by labels:

Training set

Average log-probability of a 0 image: -201.136
Average log-probability of a 1 image: -97.755
Average log-probability of a 2 image: -201.701
Average log-probability of a 3 image: -184.880
Average log-probability of a 4 image: -172.018
Average log-probability of a 5 image: -191.928
Average log-probability of a 6 image: -177.253
Average log-probability of a 7 image: -156.895
Average log-probability of a 8 image: -187.626
Average log-probability of a 9 image: -162.041

Test set

Average log-probability of a 0 image: -199.743
Average log-probability of a 1 image: -96.973
Average log-probability of a 2 image: -199.290
Average log-probability of a 3 image: -182.329
Average log-probability of a 4 image: -171.179
Average log-probability of a 5 image: -190.813
Average log-probability of a 6 image: -181.537
Average log-probability of a 7 image: -154.226
Average log-probability of a 8 image: -187.226
Average log-probability of a 9 image: -159.568

Part 3

1. Recall: $\theta_{p,q} \leftarrow \dfrac{\left(\sum_{i=1}^{N} r_p^i x_q^i\right) + a - 1}{\left(\sum_{i=1}^{N} r_p^i\right) + a + b - 2}$

If $a=b=1$, $\theta_{p,q} \leftarrow \dfrac{\sum r_p^i x_q^i}{\sum_{i=1}^{N} r_p^i}$ ← if $x_q^i = 0 \ \forall i$, $\theta_{p,q} \leftarrow 0$.

By our indep. assumption:

$$P(x \mid z=k) = \prod_{j=1}^{D} \theta_{k,j}^{x_q^i} (1-\theta_{k,j})^{x_q^i}$$

if $x_q^i = 1$, then it is clear $\forall k$, $P(x \mid z=k) = 0$.

Conceptually, based on the data, algo is 100% sure that for all classes, that pixel MUST be off. So it predicts 0 prob if by chance it is on.

2. The part 1 model only has 10 clusters for each label so it averages and misses out on variations in writing. This will be detrimental to similar looking numbers. e.g.: 7, 7, 9, 8, 9

In contrast, part 2 has 100 clusters. It can form clusters to differentiate 7, 7, 9, 8, 9

3. The average log probability for each digit class is a report of how confidently the model predicts the correct class for examples in the class by averaging over all the examples in a class.

It is clear then that the model does NOT believe 1's are more common than 8's. It means it is easier to differentiates 1's than 8's. This makes sense because 7, 9, 3 for example look like 8.

The relative frequency of sampling a class is actually given by π. Thus, from the info given, it is unknown if one would sample more 1s or 8s.

Extra

Train_from_labels output:
('Training log-likelihood:', -172.07854196938328)
('Test log-likelihood:', -170.97538133747267)

Train_with_em output:
('Final training log-likelihood:', -138.14790916090735)
('Final test log-likelihood:', -138.52928834184047)