Part 1

Question 1

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

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

$x_j | z = k \sim \text{Bernoulli}(\Theta_{k,j})$

$\Theta_{k,j} \sim \text{Beta}(a,b)$　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. $P(\pi) = \pi_1^{(c-1)}\dots\pi_k^{(c-1)}$　prior over class

How to derive ℓ above?

$\log P(\Theta,\pi,D) = \log P(D|\Theta)P(\Theta) = \log P(D|\Theta) + \log P(\Theta) + \log P(\pi)$

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

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

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

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

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

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

r_k^i is the "responsibility" has
the kth cluster
to the ith data point

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

$\underbrace{\qquad}_{r_k^i}$　Θ,π here are
constants

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

Fact:

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

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

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

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

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

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

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

$\boxed{\therefore \pi_b = \dfrac{(c-1) + \sum_{i=1}^{N}r_b^i}{K(c-1) + N}}$　　Note: as expected $\sum_{k=1}^{K}\pi_k = 1$

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

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

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

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

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

$$0 = \frac{1 - a - f}{-b + g - f - a + 2} = \Theta_{p,q}$$

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

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

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

class e^x

feature e^x

How to vectorize?

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

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

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

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

$^Pr^Kx$

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

$$P(z=k|x) = = \dfrac{\prod\limits_{j=1}^{D} \theta_{k,j}^{x_j^i}\left(1-\theta_{k,j}\right)^{1-x_j^i} \cdot \pi_k}{\sum\limits_{p=1}^{K}\left[\prod\limits_{j=1}^{D} \theta_{p,j}^{x_j^i}\left(1-\theta_{p,j}\right)^{1-x_j^i}\right]\pi_p}$$

↳ assumes fully observed though.

$Pr(z=k|X_{obs}) = \sum\limits_{X_{HIDDEN}} Pr(z=k, X_{HIDDEN}|X_{obs})$

$= \sum\limits_{X_{HIDDEN}} Pr(z=k|X_{HIDDEN}, X_{obs}) Pr(X_{HIDDEN}|X_{obs})$

↘ By independance.

$= \sum\limits_{X_{HIDDEN}} Pr(z=k|X_{HIDDEN}, X_{obs}) Pr(X_{HIDDEN})$

$= \sum\limits_{X_{HIDDEN}} \dfrac{Pr(X_{HIDDEN}, X_{obs}|z=k)Pr(z=k)}{Pr(X_{HIDDEN}, X_{obs})} \cdot Pr(X_{HIDDEN})$

↘ independence

$= \sum\limits_{X_{HIDDEN}} \dfrac{Pr(X_{HIDDEN}|z=k)Pr(X_{obs}|z=k)Pr(z=k)}{Pr(X_{HIDDEN})Pr(X_{obs})} \cdot Pr(X_{HIDDEN})$

$= \dfrac{Pr(X_{obs}|z=k)Pr(z=k)}{Pr(X_{obs})} \underbrace{\sum\limits_{X_{HIDDEN}} Pr(X_{HIDDEN}|z=k)}_{1}$

$\boxed{Pr(z=k|X_{obs}) = \dfrac{Pr(X_{obs}|z=k)\cdot Pr(z=k)}{Pr(X_{obs})}}$ — e.g. we only multiply $\theta_{k,j}$ for x_j^i observed.

Question 2

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

$$\downarrow \log$$

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

$$X = \begin{bmatrix} \text{---} x' \text{---} \\ \vdots \\ \text{---} x^N \text{---} \end{bmatrix} \qquad \Theta = \begin{bmatrix} \text{---} \Theta_1 \text{---} \\ \vdots \\ \text{---} \Theta_K \text{---} \end{bmatrix} \qquad Pr(x^2|z=j), j\in(1,K).$$

$$X\Theta^T = \begin{bmatrix} \text{---} x' \text{---} \\ \vdots \\ \text{---} x^N \text{---} \end{bmatrix} \begin{bmatrix} | & & | \\ \Theta_1 & \cdots & \Theta_K \\ | & & | \end{bmatrix} = \begin{bmatrix} x'\Theta_1 & \cdots & x'\Theta_K \\ \vdots & & \vdots \\ x^N\Theta_1 & \cdots & x^N\Theta_K \end{bmatrix} \leftarrow$$

Posterior Predictive Mean

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

How to do this?

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

$$Pr(X_j^{(i)}|z=K) = \Theta_{k,j}^{x_j} (1-\Theta_{k,j})^{1-x_j}$$
$$E[x_j^{(i)}] = (1)\Theta_{k,j} + (0)(1-\Theta_{k,j}) = \Theta_{k,j}$$

$\rightarrow$ take the argmax of posterior to get K.

Question 3

Part 2 values:
('R[0, 2]', 0.1748895149211743)
('R[1, 0]', 0.68853767610923089)
('P[0, 183]', 0.74454557952318989)
('P[2, 628]', 0.45918083205837279)

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 $x_q^i = 0 \ \forall i, \ \theta_{p,q} \leftarrow 0$.

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

   By our indep. assumption:
   $$P(x \mid z = k) = \prod_{j=1}^{D} \theta_{k,j}^{x_q^i} (1 - \theta_{k,j})^{x^i}$$

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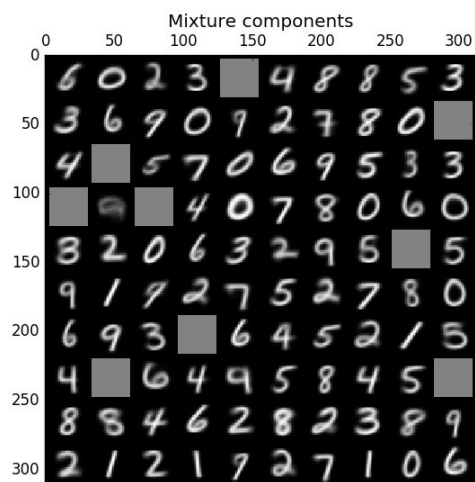('Training log-likelihood:', -172.07854196938328)
('Test log-likelihood:', -170.97538133747267)

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

EM Mixtures:

Model Predictions (top half was observed):