CSC411/2515 Fall 2018
Homework 5

Soon Chee Loong
999295793
cheeloong.soon@mail.utoronto.ca

Last Name: Soon
First Name: Chee Loong
cdf markus: soonchee

1. Gaussian Discriminant Analysis

Build a classifier to label images of handwritten digits.

Each image is 8 by 8 pixels $\in [0, 1]$ ↳ $\{0, 1, 2, \ldots, 9\}$
↳ grayscale ↳ represented as 64D vector in raster scan order

700 train, 400 test for each digit in $\{0, 1, 2, \ldots, 9\}$

Using maximum likelihood, fit a set of
10 class-conditional Gaussians with a separate, full covariance matrix for each class.

Conditional Multivariate Gaussian Probability Density

$$P(\vec{x} \mid y=k, \vec{u}, \Sigma_k) = (2\pi)^{-d/2} |\Sigma_k|^{-1/2} \exp\left(-\tfrac{1}{2}(\vec{x}-\vec{u}_k)^T \Sigma_k^{-1} (\vec{x}-\vec{u}_k)\right)$$

$$P(y=k) = \tfrac{1}{10}$$

$$\vec{\theta} = \{u_{kj}, \vec{\Sigma}_k\}, \quad k \in \{0, 1, 2, \ldots, 9\}, \quad j \in \{1, 2, \ldots, 64\}, \quad D = 64, \quad K = 10$$

$$\vec{u} = \begin{bmatrix} u_{11} & \cdots & u_{1D} \\ u_{21} & \cdots & u_{2D} \\ \vdots & \ddots & \vdots \\ u_{K1} & \cdots & u_{KD} \end{bmatrix}_{(K \times D)}, \quad \Sigma_k = \begin{bmatrix} \Sigma_{k_{11}} & \Sigma_{k_{12}} & \cdots & \Sigma_{k_{1D}} \\ \Sigma_{k_{21}} & \Sigma_{k_{22}} & \cdots & \Sigma_{k_{2D}} \\ \vdots & & \ddots & \vdots \\ \Sigma_{k_{D1}} & \cdots & & \Sigma_{k_{DD}} \end{bmatrix}_{(D \times D)}$$

Implement covariance computation yourself. (NO using np.cov(:))

Hint: To ensure numerical stability, you may choose to add a small multiple of the identity to each covariance matrix.
(Add $(0.01) \cdot I$ to each matrix).

a) Using the parameters you fit on the training set and Bayes Rule, compute the Average Conditional Log-Likelihood $= \frac{1}{N} \sum_{i=1}^{N} \log(P(y^{(i)} \mid \vec{x}^{(i)}, \theta))$ on both the train & test set and report it.

$\boxed{\begin{array}{l} \text{Train}: -0.12462 \\ \text{Test}: -0.19967 \end{array}}$

$\Rightarrow e^{-0.12462} = 0.8828$

$\Rightarrow e^{-0.19967} = 0.8214$

Generative Likelihood

$$P(\vec{x} \mid y=k, \vec{u}, \Sigma_k) = (2\pi)^{-d/2} |\Sigma_k|^{-1/2} \exp\left(-\tfrac{1}{2}(\vec{x}-\vec{u}_k)^T \Sigma_k^{-1} (\vec{x}-\vec{u}_k)\right)$$

Generative Log-Likelihood

$$\log(P(\vec{x} \mid y=k, \vec{u}, \Sigma_k))$$
$$= -\tfrac{d}{2}\log(2\pi) - \tfrac{1}{2}\log(\det(\Sigma_k)) - \tfrac{1}{2}(\vec{x}-\vec{u}_k)^T \Sigma_k^{-1} (\vec{x}-\vec{u}_k)$$
$$= -\tfrac{1}{2}\left(d \log(2\pi) + \log(\det(\Sigma_k)) + (\vec{x}-\vec{u}_k)^T \Sigma_k^{-1} (\vec{x}-\vec{u}_k)\right)$$

Conditional Likelihood

$$P(y=i \mid \vec{x}, \vec{u}, \Sigma_k)$$
$$= \frac{P(\vec{x}, y=i \mid \vec{u}, \Sigma_i)}{P(\vec{x} \mid \vec{u}, \Sigma)}$$
$$= \frac{P(\vec{x} \mid y=i, \vec{u}, \Sigma_i) P(y=i)}{\sum_{j=1}^{K} P(\vec{x} \mid y=j, \vec{u}, \Sigma_j) P(y=j)}$$

Conditional Likelihood

$P(y=i \mid \vec{x}, \vec{u}, \Sigma_k)$

$= \dfrac{P(\vec{x}, y=i \mid \vec{u}, \Sigma_i)}{P(\vec{x} \mid \vec{u}, \Sigma)}$

$= \dfrac{P(\vec{x} \mid y=i, \vec{u}, \Sigma_i)\, P(y=i)}{\sum_{j=1}^{k} P(\vec{x} \mid y=j, \vec{u}, \Sigma_j)\, P(y=j)}$

$= \dfrac{\cancel{\left(\frac{1}{10}(2\pi)^{-d/2}\right)} |\Sigma_i|^{-1/2} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{u}_i)\right)}{\left(\frac{1}{10}(2\pi)^{-d/2}\right) \sum_{j=1}^{k} |\Sigma_j|^{-1/2} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T \Sigma_j^{-1} (\vec{x}-\vec{u}_j)\right)}$

1

$= \dfrac{|\Sigma_i|^{-1/2} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{u}_i)\right)}{\sum_{j=1}^{k} |\Sigma_j|^{-1/2} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T \Sigma_j^{-1} (\vec{x}-\vec{u}_j)\right)}$

Conditional Log-Likelihood

$\log(P(y=i \mid \vec{x}, \vec{u}, \Sigma_i))$

$= \log(P(\vec{x}, y=i \mid \vec{u}, \Sigma_i)) - \log(P(\vec{x} \mid \vec{u}, \Sigma))$

$\log(P(\vec{x}, y=i \mid \vec{u}, \Sigma_i))$

$= -\log(10) - \frac{d}{2}\log(2\pi) - \frac{1}{2}\log(\det(\Sigma_i)) - \frac{1}{2}(\vec{x}-\vec{u}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{u}_i)$

$\log(P(\vec{x} \mid \vec{u}, \Sigma))$

$= -\log(10) - \frac{d}{2}\log(2\pi) + \log\left(\sum_{j=1}^{k} \det(\Sigma_j)^{-1/2} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T \Sigma_j^{-1} (\vec{x}-\vec{u}_j)\right)\right)$

$= -\log(10) - \frac{d}{2}\log(2\pi) + \log\left(\sum_{j=1}^{k} \det(\Sigma_j)^{-1/2}\right)$

$+ \log\left(\sum_{j=1}^{k} \exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T \Sigma_j^{-1} (\vec{x}-\vec{u}_j)\right)\right)$

scipy.misc.logsumexp

$\dfrac{\exp(A_i)}{\sum_j \exp(A_j)} = \dfrac{\exp(A_i - B)}{\sum_j \exp(A_j - B)} = \dfrac{\exp(A_i - \max_j(A_j))}{\sum_j \exp(A_j - \max_j(A_j))} \rightarrow$ more numerically stable

$\sum_j \exp(A_j) = \sum_j \exp(A_j)\exp(B)\exp(-B) = \sum_j \exp(A_j - B)\exp(B)$

$= \exp(B) \sum_j \exp(A_j - B)$

$= \exp(\max_i(A_i)) \sum_j \exp(A_j - \max_i(A_i))$

Let $A_j = -\frac{1}{2}(\vec{x}-\vec{u}_j)^T \Sigma_j^{-1} (\vec{x}-\vec{u}_j)$

$A = \begin{pmatrix} A_1 \\ \vdots \\ A_k \end{pmatrix}$

scipy.misc.logsumexp(A)

$\log(P(y=i \mid \vec{x}, \vec{u}, \Sigma_i))$

$= \log(P(\vec{x}, y=i \mid \vec{u}, \Sigma_i)) - \log(P(\vec{x} \mid \vec{u}, \Sigma))$

scipy.misc.logsumexp(A)

$\log(P(y=i|\vec{x},\vec{u},\Sigma_i))$

$= \log(P(\vec{x},y=i|\vec{u},\Sigma_i)) - \log(P(\vec{x}|\vec{u},\Sigma))$

$= -\frac{1}{2}\log(\det(\Sigma_i)) - \frac{1}{2}(\vec{x}-\vec{u}_i)^T\Sigma_i^{-1}(\vec{x}-\vec{u}_i)$

$-\left(\log\left(\sum_{j=1}^{K}\det(\Sigma_j)^{-1/2}\right) + \log\left(\sum_{j=1}^{K}\exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T\Sigma_j^{-1}(\vec{x}-\vec{u}_j)\right)\right)\right)$

BUT want to re-use earlier code.

$\log(P(y=i|\vec{x},\vec{u},\Sigma_i))$

$= \log(P(\vec{x},y=i|\vec{u},\Sigma_i)) - \log(P(\vec{x}|\vec{u},\Sigma))$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) + \log(P(y=i))$

$- \log\left(\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\,P(y=j)\right)$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) + \log(P(y))$, since $P(y=i)=P(y)=\frac{1}{10}=P(y=j)\ \forall i,j$

$- \log\left(\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\quad P(y)\right)$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) + \log(P(y))$

$- \log\left(P(y)\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\right)$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) + \cancel{\log(P(y))}$

$- \log\left(\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\right) - \cancel{\log(P(y))}$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) - \log\left(\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\right)$

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) - \log\left(\sum_{j=1}^{K}(P(\vec{x}|y=j|\vec{u},\Sigma_j)\right)$

↳ computed before ↳ denominator ⇒ doesn't affect prediction

$= \log(P(\vec{x}|y=i|\vec{u},\Sigma_i)) + \frac{d}{2}\log(2\pi) - \log\left(\sum_{j=1}^{K}\det(\Sigma_j)^{-1/2}\right)$

$- \log\left(\sum_{j=1}^{K}\exp\left(-\frac{1}{2}(\vec{x}-\vec{u}_j)^T\Sigma_j^{-1}(\vec{x}-\vec{u}_j)\right)\right)$

↳ use logsumexp() here

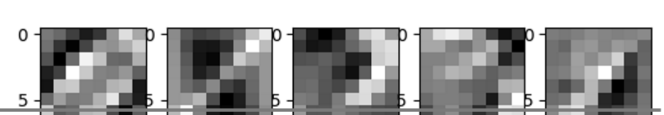
Average Conditional Log-Likelihood $=\frac{1}{N}\sum_{i=1}^{N}\log(P(y^{(i)}|\vec{x}^{(i)},\theta))$

b) Select the most likely posterior class for each training & test data point as your prediction. Report accuracy on the train and test set.

Train: 0.98142857 → 98%

Test: 0.97275 ⇒ 97%

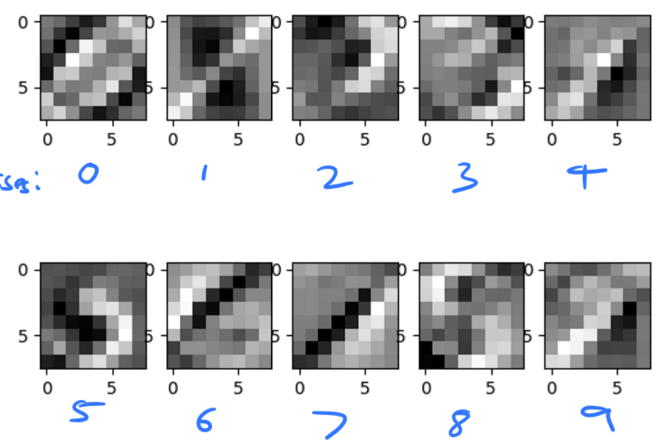
c) Compute the leading eigenvector (largest eigenvalue) for each class covariance matrix and plot them side by side as 8 by 8 images. Hint: Use np.linalg.eig



b) Select the most likely posterior class for each training & test data point as your prediction. Report accuracy on the train and test set.

Train: 0.9814285 ⇒ 98%
Test: 0.97275 ⇒ 97%

c) Compute the leading eigenvector (largest eigenvalue) for each class covariance matrix and plot them side by side as 8 by 8 images. Hint: Use np.linalg.eig



Code submitted as q1.py

```
root@soon:~/Github/CSC411Fall2018Assignments/Homework5# bash runAll.sh
Train data shape:  (7000, 64)
Train labels shape:  (7000,)
Test data shape:  (4000, 64)
Test labels shape:  (4000,)
Train Average Conditional Log Likelihood: -0.12462443666863064
Test Average Conditional Log Likelihood: -0.1966732032552559
Train Average Conditional Likelihood: 0.8828283983061755
Test Average Conditional Likelihood: 0.8214590395931995
Train Accuracy:  0.9814285714285714
Test Accuracy:  0.97275
root@soon:~/Github/CSC411Fall2018Assignments/Homework5#
```

Run Output of Code.
I also converted log likelihood to probability.

2. Categorical Distribution ⇒ Dirichlet Distribution with $\alpha_i = 1$ for $i \in \{1, 2, \ldots, K\}$

↳ discrete distribution over K outcomes, $\{1, 2, \ldots, K\}$

θ_k = Probability of outcome for category k

$\theta_k \geq 0$

$\sum_{k=1}^{K} \theta_k = 1$

$\underset{(K \times 1)}{\vec{x}}$ = observation with 1-of-K encoding ⇒ one entry is 1, other (K−1) entries are 0

$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{bmatrix}$

Probability of an Observation

$P(\vec{x}; \vec{\theta}) = \prod_{k=1}^{K} \theta_k^{x_k}$

N_k = count for outcome k

N = total # of observations

2. Categorical Distribution ⇒ Dirichlet Distribution with $\alpha_i = 1$ for $i \in \{1, 2, \dots, K\}$

↳ discrete distribution over K outcomes, $\{1, 2, \dots, K\}$

θ_k = Probability of outcome for category k

$\theta_k \geq 0$

$\sum_{k=1}^{K} \theta_k = 1$

$\vec{x}_{(K\times1)}$ = observation with 1-of-K encoding ⇒ one entry is 1, other $(K-1)$ entries are 0

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{bmatrix}$$

Probability of an Observation

$P(\vec{x}; \vec{\theta}) = \prod_{k=1}^{K} \theta_k^{x_k}$

N_k = count for outcome k

N = total # of observations

$\Rightarrow \hat{\theta}_{k_{M.L.E.}} = \frac{N_k}{N}$ as derived in previous assignment. (Given as fact in assignment)

Derive the Bayesian parameter estimate.

Use the Dirichlet distribution for the prior.

$P(\vec{\theta}) \propto \theta_1^{a_1 - 1} \cdots \theta_k^{a_k - 1}$

$\vec{\theta} \sim \text{Dirichlet}(a_1, \dots, a_k) \Rightarrow E[\theta_k] = \frac{a_k}{\sum_{k'} a_{k'}}$ (Given as fact in assignment)

a) i) Determine the posterior distribution $P(\vec{\theta} \mid D)$ where D is the set of observations.

Let $X_{(N\times K)} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_K^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_K^{(2)} \\ \vdots & & \ddots & \vdots \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_K^{(N)} \end{bmatrix} = \begin{bmatrix} \vec{x}^{(1)T} \\ \vec{x}^{(2)T} \\ \vdots \\ \vec{x}^{(N)T} \end{bmatrix}$, $\vec{x}^{(i)}$ = i^{th} observation

Posterior Distribution

$P(\vec{\theta} \mid D)$

$\propto P(D \mid \vec{\theta}) P(\vec{\theta})$, since $P(D)$ is constant

$= \left(\prod_{i=1}^{N} \prod_{j=1}^{K} \theta_j^{x_j^{(i)}} \right) P(\vec{\theta})$

$= \left(\prod_{j=1}^{K} \theta_j^{N_j} \right) P(\vec{\theta})$

$\propto \left(\prod_{j=1}^{K} \theta_j^{N_j} \right) \prod_{j=1}^{K} \theta_j^{a_j - 1}$

$= \prod_{j=1}^{K} \theta_j^{N_j + a_j - 1} \Rightarrow \text{Dirichlet}(N_1 + a_1, \dots, N_K + a_K)$

∴, proven posterior distribution is also a Dirichlet distribution

$$\boxed{P(\vec{\theta} \mid D) = \text{Dirichlet}(\alpha_1 + N_1, \dots, \alpha_K + N_K)}$$

a) ii) Then, determine the posterior predictive probability that the next outcome will be k.

Since given $\vec{\theta} \sim \text{Dirichlet}(a_1, \dots, a_k) \Rightarrow E[\theta_k] = \frac{a_k}{\sum_{k'} a_{k'}}$

$\Rightarrow \vec{\theta} \mid D \sim \text{Dirichlet}(\alpha_1 + N_1, \dots, \alpha_K + N_K) \Rightarrow E[\theta_k \mid D] = \frac{\alpha_k + N_k}{\sum_{k'} \alpha_{k'} + N_{k'}}$

Posterior Predictive Probability

$P(x = i \mid D) = \int P(x = i \mid \vec{\theta}) P(\vec{\theta} \mid D) d\vec{\theta}$

$= \int \theta_i P(\vec{\theta} \mid D) d\vec{\theta}$

a) ii) Then, determine the posterior predictive probability that the next outcome will be k.

Since given $\vec{\theta} \sim \text{Dirichlet}(a_1, \dots, a_k) \Rightarrow E[\theta_k] = \frac{a_k}{\sum_{k'} a_{k'}}$

$\Rightarrow \vec{\theta} \mid D \sim \text{Dirichlet}(\alpha_1 + N_1, \dots, \alpha_k + N_k) \Rightarrow E[\vec{\theta}_k \mid D] = \frac{\alpha_k + N_k}{\sum_{k'} \alpha_{k'} + N_{k'}}$

Posterior Predictive Probability

$$P(x = i \mid D) = \int P(x = i \mid \vec{\theta}) P(\vec{\theta} \mid D) d\vec{\theta}$$
$$= \int \theta_i P(\vec{\theta} \mid D) d\vec{\theta}$$
$$= E[\theta_i \mid D] \text{, just expectation for random variable } \theta_i$$
$$= \frac{\alpha_i + N_i}{\sum_{j=1}^{K} (\alpha_j + N_j)} \text{, from}$$

$\therefore$, proven.

$$\boxed{P(x = i \mid D) = \frac{\alpha_i + N_i}{\sum_{j=1}^{K} (\alpha_j + N_j)}}$$

b) Determine the MAP estimate of the parameter vector $\vec{\theta}$.

May assume each $a_k > 1$

$$\hat{\theta}_{M.A.P.} = \underset{\theta}{\arg\max} (P(\vec{\theta} \mid D))$$

$\vec{\theta} \mid D \sim \text{Dirichlet}(\alpha_1 + N_1, \dots, \alpha_k + N_k)$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{\alpha_j + N_j - 1}$

Let $N_k' = N_k + \alpha_k - 1$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{N_j + \alpha_j - 1} = \prod_{j=1}^{K} \theta_j^{N_j'}$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{N_j'} \Rightarrow$ A categorical distribution

Since given $\hat{\theta}_{i\,M.L.E.}' = \frac{N_i'}{\sum_{j=1}^{K} N_j'}$ for categorical distribution,

$$\Rightarrow \hat{\theta}_{i\,M.L.E.}' = \hat{\theta}_{i\,M.A.P.} = \frac{N_i'}{\sum_{j=1}^{K} N_j'} = \frac{N_i + \alpha_i - 1}{\sum_{j=1}^{K} N_j + \alpha_j - 1}$$

$\therefore$, proven

$$\hat{\theta}_{M.A.P.} = \begin{bmatrix} \hat{\theta}_{1\,M.A.P.} \\ \hat{\theta}_{2\,M.A.P.} \\ \vdots \\ \hat{\theta}_{k\,M.A.P.} \end{bmatrix}, \quad \boxed{\hat{\theta}_{i\,M.A.P.} = \frac{N_i + \alpha_i - 1}{\sum_{j=1}^{K} N_j + \alpha_j - 1} = \frac{N_i + \alpha_i - 1}{N - K + \sum_{j=1}^{K} \alpha_j}}$$

$$= \frac{N_i + \alpha_i - 1}{\underbrace{\sum_{j=1}^{K} N_j}_{N} + \underbrace{\sum_{j=1}^{K} (-1)}_{-K} + \sum_{j=1}^{K} \alpha_j}$$

$, \alpha_i > 1 \ \forall i$
$i \in \{1, 2, \dots, K\}$

b) Determine the MAP estimate of the parameter vector $\vec{\theta}$.
May assume each $a_k > 1$

$$\hat{\theta}_{M.A.P.} = \arg\max_{\theta} (P(\vec{\theta} \mid D))$$

$\vec{\theta} \mid D \sim \text{Dirichlet}(\alpha_1 + N_1, \ldots, \alpha_K + N_K)$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{\alpha_j + N_j - 1}$

Let $N_k' = N_k + \alpha_k - 1$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{N_j + \alpha_j - 1} = \prod_{j=1}^{K} \theta_j^{N_j'}$

$\Rightarrow P(\vec{\theta} \mid D) \propto \prod_{j=1}^{K} \theta_j^{N_j'} \Rightarrow$ A categorical distribution

Since given $\hat{\theta}_{i\,M.L.E.}' = \dfrac{N_i'}{\sum_{j=1}^{K} N_j'}$ for categorical distribution,

$$\Rightarrow \hat{\theta}_{i\,M.L.E.}' = \hat{\theta}_{i\,M.A.P.} = \frac{N_i'}{\sum_{j=1}^{K} N_j'} = \frac{N_i + \alpha_i - 1}{\sum_{j=1}^{K} N_j + \alpha_j - 1}$$

$\therefore$, proven

$$\hat{\theta}_{M.A.P.} = \begin{bmatrix} \hat{\theta}_{1\,M.A.P.} \\ \hat{\theta}_{2\,M.A.P.} \\ \vdots \\ \hat{\theta}_{K\,M.A.P.} \end{bmatrix}, \quad \boxed{\hat{\theta}_{i\,M.A.P.} = \frac{N_i + \alpha_i - 1}{\sum_{j=1}^{K} N_j + \alpha_j - 1} = \frac{N_i + \alpha_i - 1}{N - K + \sum_{j=1}^{K} \alpha_j}}$$

$$= \frac{N_i + \alpha_i - 1}{\underbrace{\sum_{j=1}^{K} N_j}_{N} + \underbrace{\sum_{j=1}^{K} (-1)}_{-K} + \sum_{j=1}^{K} \alpha_j}$$

$, \alpha_i > 1 \ \forall i$
$i \in \{1, 2, \ldots, K\}$

To check answer:
note: if $\alpha_j = 1 \ \forall j \in \{1, 2, \ldots, K\}$

$\Rightarrow \hat{\theta}_{i\,M.A.P.} = \dfrac{N_i + (1-1)}{\sum_{j=1}^{K} N_j + (1-1)} = \dfrac{N_i}{N}$ $\Rightarrow$ same as maximum likelihood estimation for Multinomial distribution.