REPORT - PROJECT 1 (Dharmit Prajapati)

1) For task 1:

we have a 28x28 array of pixels per each image

For python code,

we have flattened each image ~~into~~ from 28x28 to 1x784

For each image we have found out mean (m_i) & standard deviation ~~sd~~ (s_i)

∴ Each image is now represent as [m_i, s_i]

Then we calculate

M1 = mean of all means of all images

M2 = mean ~~of~~ of all SDs of all images

S1 = SD of all means of all images

S2 = SD of all SDs of all images

∴ Finally each image is now represented by normalized feature vector Y_i

$$Y_i = [y_{1i}, y_{2i}] \overset{t}{=} [(m_i - M_1)/S_1, (s_i - M_2)/S_2]^t$$

∴ Thus, we have done feature extraction & normalization which are one of the initial stages of Pattern recognition

2) Task 2 represents the more general case of estimating $\hat{\theta}$ where μ and Σ are both unknowns. (For gaussian/normal distribution)

Let θ_1 & θ_2 be the unknown parameters that constitute components of parameter vector θ.

Considering it for univariate case with $\theta_1 = \mu$ & $\theta_2 = \sigma^2$

we know,

$$p(x_k \mid \theta) = \frac{1}{\sqrt{2\pi\theta_2}} \exp\left[\frac{-1}{2\theta_2}(x_k-\theta_1)^2\right]$$

$$\ln p(x_k \mid \theta) = \frac{-1}{2}\ln 2\pi\theta_2 - \frac{1}{2\theta_2}(x_k-\theta_1)^2$$

$$\nabla_\theta l = \nabla_\theta \ln(p(x_k/\theta)) = \begin{bmatrix} \frac{1}{\theta_2}(x_k-\theta_1) \\ -\frac{1}{2\theta_2} + \frac{(x_k-\theta_1)^2}{2\theta_2^2} \end{bmatrix}$$ (Taking partial derivate wrt θ_1 & θ_2)

we know,

$$\nabla_\theta l = \sum_{k=1}^{n} \nabla_\theta \ln(p(x_k/\theta)) = 0$$ (for estimating $\hat{\theta}$)

$$\therefore \sum_{k=1}^{n} \frac{1}{\theta_2}(x_k-\theta_1) = 0$$

$$\sum_{k=1}^{n} (x_k - \theta_1) = 0$$

$$\sum_{k=1}^{n} x_k = \theta_1(n)$$

$$\therefore \boxed{\theta_1 = \frac{1}{n}\sum_{k=1}^{n} x_k} \quad \boxed{\text{ie } \hat{\theta}_1 = \mu}$$

also $$\sum_{k=1}^{n} \frac{-1}{2\theta_2} + \sum_{k=1}^{n} \frac{(x_k-\theta_1)^2}{2\theta_2^2} = 0$$

$$\boxed{\theta_2 = \frac{1}{n}\sum_{k=1}^{n}(x_k-\theta_1)^2}$$

ie $$\boxed{\hat{\theta}_2 = \sigma_2}$$

For a multivariate case,

we can again prove that the parameters are mean & covariance

$\therefore$ In the Task 2 the ~~normal~~ parameters for 2-D normal distribution will be sample mean & covariance matrix respectively.

ie $\hat{\mu} = \frac{1}{n}\sum_{k=1}^{n} x_k$

$$\hat{\Sigma} = \frac{1}{n}\sum_{k=1}^{n} (x_k - \hat{\mu})(x_k - \hat{\mu})^t$$

3) For task 3, in order to obtain the probability of error we use the P(error) function

$$P(error) = \int_{-\infty}^{\infty} P(error|x)\, p(x)\, dx$$

we choose class 3 over class 7 (let w_1 = class3 & w_2 = class 7):

$$P(w_1|x) > P(w_2|x)$$

ie $p(x|w_1) \cdot P(w_1) > p(x|w_2) \cdot P(w_2)$

we know,

~~P(error)~~ $P(error|x) = \min(P(w_1|x), P(w_2|x))$ - ①

$$= \min\left(\frac{p(x|w_1) \cdot P(w_1)}{P(x)}, \frac{p(x|w_2)\, P(w_2)}{p(x)}\right)$$

$\therefore$ $P(error) = \int_{-\infty}^{\infty} \min(p(x|w_1) \cdot P(w_1), p(x|w_2) \cdot P(w_2))\, dx$ - ②

In the python code,

we have summed up all the error according to ① & ②. and divided by the number of samples.

```
[Command: python -u D:\Dharmit\
FSL_task1.py']
[M1 M2]: [32.50435107 76.44042397]
[S1 S2]: [ 9.34944945 10.50972539]
Mu3(Mean3) [0.37687996 0.31851855]
Mu7(Mean7) [-0.36900004 -0.31185886]
Sigma3 [[1.0491056  0.98717364]
 [0.98717364 0.96037982]]
Sigma7 [[0.67669136 0.74435619]
 [0.74435619 0.842203  ]]
Total error for training set for case1 is: 0.1584498396653004
Total error for training set for case2 is: 0.1053831567729471l
Total error for test set for case1 is: 0.1537071866992889б
Total error for test set for case2 is: 0.1022330369451785б
[Finished in 7.455s]
```