Homework 3

Due: Monday, Oct-12-2020, 5pm – Gradescope entry code: 948XVG

Please upload your answers timely to Gradescope. Start a new page for every problem. For the programming/simulation questions you can use any reasonable programming language. Comment your source code and include the code and a brief overall explanation with your answers.

1. Exercise 3.4 in text.

Solution:

(a) Let $Z = X_1 + X_2$. Since $X_1$ and $X_2$ are independent, the density of $Z$ can be calculated as

$$f_Z(z) = \int_{-\infty}^{\infty} f_{X_1}(x) f_{X_2}(z-x \mid X_1 = x) dx$$

$$= \int_{-\infty}^{\infty} f_{X_1}(x) f_{X_2}(z-x) dx$$

$$= (f_{X_1} * f_{X_2})(z)$$

which is the convolution of the densities of $X_1$ and $X_2$. Then

$$f_Z(z) = \int_{-\infty}^{\infty} f_{X_1}(x) f_{X_2}(z-x) dx = \frac{1}{2\pi \sigma_1 \sigma_2} \int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma_1^2}} e^{-\frac{(z-x)^2}{2\sigma_2^2}} dx$$

$$= \frac{1}{2\pi \sigma_1 \sigma_2} \exp\left(-\frac{z^2}{2\sigma_2^2} + \frac{z^2}{(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2})}\right) \int_{-\infty}^{\infty} \exp\left(-\frac{x^2}{2\sigma_1^2} + \frac{x^2}{2(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2})}\right) dx$$

$$= \frac{1}{2\pi \sigma_1 \sigma_2} \exp\left(-\frac{z^2}{2\sigma_2^2} + \frac{z^2}{(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2})}\right) \sqrt{2\pi} \sqrt{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \int_{-\infty}^{\infty} e^{-\frac{z^2}{2\sigma_2^2} + \frac{z^2}{(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2})}} dx$$

$$= \frac{1}{\sqrt{2\pi} \sqrt{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}} \exp\left(-\frac{z^2}{2\sigma_2^2} + \frac{z^2}{2(\sigma_2^2 + \frac{\sigma_1^2}{\sigma_2^2})}\right)$$

$$= \frac{1}{\sqrt{2\pi} \sqrt{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}} \exp\left(-\frac{z^2}{2(\sigma_1^2 + \sigma_2^2)}\right)$$,

(1)
where (1) follows since
\[ \sqrt{\frac{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}{2\pi}} e^{-\frac{1}{2} \left( \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2} \right) \left( x - \frac{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \right)^2} \]
is the pdf of \( \mathcal{N} \left( \frac{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}}{\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}} \right) \) and hence integrates to 1. This completes the proof that \( X_1 + X_2 \sim \mathcal{N} \left( 0, \sigma_1^2 + \sigma_2^2 \right) \).

Note: It is also a valid approach to use the Fourier transform or the moment generating function to perform the convolution (provided you have calculated the transform, and justified the relationship in the transform domain and the uniqueness of the transform).

(b) Follow the hint by setting \( \sigma_X^2 = a_i^2 \) for \( i = 1, 2 \). Thus, \( X_i \sim \mathcal{N} \left( 0, \sigma_X^2 \right) \). Use the results in Part (a) to derive
\[ aW_1 + bW_2 \sim \mathcal{N} \left( 0, a_1^2 + a_2^2 \right). \]

(c) The inductive hypothesis is that if \( \{W_i; i \geq 1\} \) is a sequence of i.i.d. standard Gaussian rv's, if \( \{a_i; i \geq 1\} \) is a sequence of numbers, and if \( \sum_{i=1}^n a_i W_i \sim \mathcal{N} \left( 0, \sum_{i=1}^n a_i^2 \right) \) for a given \( n \geq 1 \), then \( \sum_{i=1}^{n+1} a_i W_i \sim \mathcal{N} \left( 0, \sum_{i=1}^{n+1} a_i^2 \right) \).

The basis for this induction \((n = 2)\) was established in Part (b). For the inductive step, let \( X = \sum_{i=1}^n a_i W_i \), then applying (b) yields
\[ X + a_{n+1} W_{n+1} \sim \mathcal{N} \left( 0, \sigma_X^2 + a_{n+1}^2 \right) = \mathcal{N} \left( 0, \sum_{i=1}^{n+1} a_i^2 \right). \]

This establishes the inductive step, so \( \sum_{i=1}^n a_i W_i \sim \mathcal{N} \left( 0, \sum_{i=1}^n a_i^2 \right) \) for all integer \( n \geq 1 \), i.e. all linear combinations of i.i.d. standard Gaussian rv's are Gaussian rv's.

2. Exercise 3.9 in text.

Solution:
The covariance matrix of the vector \([X, Y]^T\) is
\[ K = \begin{bmatrix} \sigma_X^2 & \rho \sigma_X \sigma_Y \\ \rho \sigma_X \sigma_Y & \sigma_Y^2 \end{bmatrix}. \]

One can compute
\[ \det (K) = (1 - \rho^2) \sigma_X^2 \sigma_Y^2, \]
and
\[ K^{-1} = \frac{1}{(1 - \rho^2) \sigma_X^2 \sigma_Y^2} \begin{bmatrix} \sigma_Y^2 & -\rho \sigma_X \sigma_Y \\ -\rho \sigma_X \sigma_Y & \sigma_X^2 \end{bmatrix}. \]
The joint distribution is then given by

\[ f_{X,Y}(x,y) = \frac{1}{2\pi \sqrt{\det(K)}} \exp \left( -\frac{[x-m_X,y-m_Y]^T K^{-1} \begin{bmatrix} x-m_X \\ y-m_Y \end{bmatrix}}{2} \right) \]

\[ = \frac{1}{2\pi \sqrt{(1-\rho^2)\sigma_X^2 \sigma_Y^2}} \exp \left( -\frac{[x-m_X,y-m_Y]^T K^{-1} \begin{bmatrix} x-m_X \\ y-m_Y \end{bmatrix}}{2} \right). \]

Besides, the marginal distribution of \( Y \) is

\[ f_Y(y) = \frac{1}{\sqrt{2\pi \sigma_Y}} \exp \left( -\frac{(y-m_Y)^2}{2\sigma_Y^2} \right). \]

One can now compute the conditional distribution as follows

\[ f_{X|Y}(x|y) = \frac{f_{X,Y}(x,y)}{f_Y(y)} \]

\[ = \frac{1}{2\pi \sqrt{(1-\rho^2)\sigma_X^2 \sigma_Y^2}} \exp \left( -\frac{[x-m_X,y-m_Y]^T K^{-1} \begin{bmatrix} x-m_X \\ y-m_Y \end{bmatrix}}{2} \right) \]

\[ = \frac{1}{\sqrt{2\pi \sqrt{(1-\rho^2)\sigma_X^2}}} \exp \left( -\frac{(y-m_Y)^2}{2\sigma_Y^2} \right) \]

\[ = \frac{1}{\sqrt{2\pi \sqrt{(1-\rho^2)\sigma_X^2}}} \exp \left( -\frac{\frac{1}{\sigma_X^2}(x-m_X)^2 - \frac{2\rho}{\sigma_X \sigma_Y} (x-m_X)(y-m_Y) + \frac{\rho^2}{\sigma_Y^2} (y-m_Y)^2}{2(1-\rho^2)} \right) \]

\[ = \frac{1}{\sqrt{2\pi \sqrt{(1-\rho^2)\sigma_X^2}}} \exp \left( -\frac{(x-m_X - \frac{\rho \sigma_X}{\sigma_Y} (y-m_Y))^2}{2\sigma_X^2 (1-\rho^2)} \right), \]

which indicates that \( X|Y = y \sim N \left( m_X + \frac{\rho \sigma_X}{\sigma_Y} \ (y-m_Y), \sigma_X^2 \ (1-\rho^2) \right). \)

3. Exercise 3.17 in text.

**Solution:**

(a) Let \( a \) be an arbitrary real \( n \)-vector and \( b \) be an arbitrary real \( m \) vector. Then \( a^T X \) is a Gaussian rv. Also, \( b^T Z \) is a Gaussian rv and is independent of \( a^T X \).
Consequently, $a^\top X + b^\top Z$ is a Gaussian rv. Since $a$ and $b$ are arbitrary, this shows that all linear combinations of $X_1, \ldots, X_n, Z_1, \ldots, Z_m$ are Gaussian, and thus that $X_1, \ldots, X_n, Z_1, \ldots, Z_m$ are jointly Gaussian. In the same way, any linear combination $a^\top X + b^\top Y = a^\top X + b^\top HX + b^\top Z$ is the sum of two independent rv’s, $(a^\top + b^\top H) X$ and $b^\top Z$. Thus, $X_1, \ldots, X_n, Y_1, \ldots, Y_m$ are jointly Gaussian.

Alternatively, we can write $X = AW_1$ and $Z = BW_2$ for some matrices $A, B$ and independent IID normal vectors $W_1, W_2$. Hence,

$$\begin{bmatrix} X \\ Z \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}$$

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} A & 0 \\ HA & B \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}$$

where $\begin{bmatrix} W_1 \\ W_2 \end{bmatrix}$ is also a IID normal vector. Hence $X_1, \ldots, X_n, Z_1, \ldots, Z_m$ are jointly Gaussian, and $X_1, \ldots, X_n, Y_1, \ldots, Y_m$ are jointly Gaussian.

(b) For any vectors $a$ and $b$, since $X$ and $Z$ are statistically independent, one can compute

$$\text{Var} \left( \begin{bmatrix} a^\top, b^\top \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \right) = \text{Var} \left( a^\top X + b^\top HX + b^\top Z \right)$$

$$= \text{Var} \left( (a + H^\top b)^\top X + b^\top Z \right)$$

$$= \text{Var} \left( (a + H^\top b)^\top X \right) + \text{Var} \left( b^\top Z \right)$$

$$= \left( a + H^\top b \right)^\top K_X \left( a + H^\top b \right) + b^\top K_Z b.$$ 

If $b \neq 0$, then one has

$$\text{Var} \left( \begin{bmatrix} a^\top, b^\top \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \right) \geq \text{Var} [b^\top Z] = b^\top K_Z b > 0$$

since $K_Z$ is non-singular. If $b = 0$ and $a \neq 0$, then

$$\text{Var} \left( \begin{bmatrix} a^\top, b^\top \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \right) = a^\top K_X a > 0$$

since $K_X$ is non-singular. In summary, for any $[a^\top, b^\top] \neq 0$, $[a^\top, b^\top] \begin{bmatrix} X \\ Y \end{bmatrix}$ is a Gaussian rv with non-zero variance, and hence $K$ is non-singular.

4. Let $X$ and $Y$ be a $n$-dimensional random vector and a $m$-dimensional random vector respectively. Define $K_{XY} := E[XY^T]$ as the cross-covariance matrix between $X$ and $Y$. 

4
a) What is the dimension of $K_{XY}$? Is it square? Is it symmetric? Is $K_{XY} = K_{YX}$?

b) Suppose $m = n$ and let $Z = X + Y$. Compute the covariance of $K_Z$ in terms of $K_X, K_Y$ and $K_{XY}$. When is $K_Z = K_X + K_Y$?

c) For general $m,n$ and $k$, let $A$ be a $k$ by $n$ matrix and $B$ be a $k$ by $m$ matrix. Compute the covariance of $Z = AX + BY$.

Solution:

a) $XY^T$ is a $n \times m$ matrix, hence $K_{XY}$ is also $n \times m$. It is not square unless $m = n$. It is not symmetric since $XY^T \neq YX^T$. For the same reason, $K_{XY} \neq K_{YX}$ (however, $K_{XY} = K_{YX}^T$).

b) Assume $X$ and $Y$ are zero-mean. Then


If $X$ and $Y$ are not zero-mean, you only need to replace $X$ by $X - \mu_X$ and $Y$ by $Y - \mu_Y$ in the definition of $K_{XY}$.

We have $K_Z = K_X + K_Y$ if and only if $K_{XY} = 0$, i.e. $X$ and $Y$ are uncorrelated. (This means that each element of $X$ is uncorrelated with each element of $Y$).

c) Similarly in this case,

$$K_Z = E[(AX + BY)(AX + BY)^T] = E[AX X^T A^T] + E[AX Y^T B^T] + E[BY X^T A^T] + E[BY Y^T B^T] = AK_X A^T + AK_{XY} B^T + BK_{YX} A^T + BK_Y B^T = AK_X A^T + AK_{XY} B^T + BK_{YX} A^T + BK_Y B^T$$

For any $m,n,k$, the result is a $k \times k$ matrix.

5. Exercise 8.1 in text.

Solution:

a) Taking the expectation of (8.40) conditional on $X = a$ yields

$$E[LLR(Y) \mid X = a] = \frac{(b - a)^T}{\sigma^2} E \left[ Y - \frac{b + a}{2} \mid X = a \right] = \frac{(b - a)^T}{\sigma^2} \left[ a - \frac{b + a}{2} \right] = \frac{(b - a)^T (b - a)}{2\sigma^2}.$$
b) The result in (a) can be expressed as \( \mathbb{E}[\text{LLR}(Y) \mid X = a] = -\frac{\|b - a\|^2}{2\sigma^2} \), from which the result follows.

c) Using the hint

\[
\text{Var}[\text{LLR}(Y) \mid X = a] = \frac{(b - a)^\top \mathbb{E}[ZZ^\top]}{\sigma^2} \frac{(b - a)}{\sigma^2} = \frac{\|b - a\|^2}{\sigma^2},
\]

from which the result follows.

d) Conditional on \( X = a \), we see that \( Y = a + Z \) is Gaussian and thus \( \text{LLR}(Y) \) is also Gaussian conditional on \( X = a \). Using the condition mean and variance of \( \text{LLR}(Y) \) found in (b) and (c),

\[
\text{LLR}(Y) \sim \mathcal{N}(-2\gamma^2, 4\gamma^2).
\]

When the rv \( \text{LLR}(Y) \) is divided by \( 2\gamma \), the conditional mean is also divided by \( 2\gamma \), and the variance is divided by \( (2\gamma)^2 \), leading to the desired result.

e) The first equality above is simply the result of a threshold test with the threshold \( \eta \). The second uses the fact in (d) that \( \text{LLR}(Y)/2\gamma \), conditional on \( a \), is \( \mathcal{N}(-\gamma, 1) \). This is a unit variance Gaussian rv with mean \(-\gamma \). The probability that it exceeds \( \ln \eta/2\gamma \) is then \( Q(\ln \eta/2\gamma + \gamma) \).

f) One can simply rewrite each equation above, but care is needed in observing that the likelihood ratio requires a convention for which hypothesis goes on top of the fraction. Thus, here the sign of the LLR is opposite to that in parts (a) to (e). This also means that the error event occurs on the opposite side of the threshold.

6. Exercise 8.9 in text.

Solution:

(a) This is simply a case of binary detection with an additive Gaussian noise rv. To prevent simply copying the answer from Example 8.2.3, the signal \( a \) associated with \( X = 0 \) is \( a = 5 \) and the signal \( b \) associated with \( X = 1 \) is \( b = 1 \). Thus \( b < a \), contrary to the assumption in Example 8.2.3. Looking at that example, we see that (8.27), repeated below, is still valid.

\[
\text{LLR}(y) = \left[ \left( \frac{b - a}{\sigma^2} \right) \left( y - \frac{b + a}{2} \right) \right] \begin{cases} \geq \hat{\delta}(y) = 1 & \text{if } \leq \hat{\delta}(y) = 0 \text{ in } \ln(\eta) . \end{cases}
\]

We can get a threshold test on \( y \) directly by first taking the negative of this expression and then dividing both sides by the positive term \( (a - b)/\sigma^2 \) to get

\[
\begin{align*}
y \leq \hat{\delta}(y) = 1 & \quad \frac{-\sigma^2 \ln(\eta)}{a - b} + \frac{b + a}{2} = -\frac{\sigma^2 \ln(\eta)}{4} + 3.
\end{align*}
\]
We get the same equation by switching the association of $X = 1$ and $X = 0$, which also changes the sign of the log threshold. The probability of error is then given by

$$\Pr \{ e_\eta \mid X = 0 \} = \Pr \left\{ Z \leq -\frac{\sigma^2 \ln(\eta)}{a - b} + \frac{b - a}{2} \right\} = Q \left( \frac{\sigma \ln(\eta)}{4} + \frac{2}{\sigma} \right)$$

$$\Pr \{ e_\eta \mid X = 1 \} = \Pr \left\{ Z > -\frac{\sigma^2 \ln(\eta)}{a - b} + \frac{a - b}{2} \right\} = Q \left( \frac{-\sigma \ln(\eta)}{4} + \frac{2}{\sigma} \right)$$

For the MAP case, set $\eta = \frac{p_0}{1 - p_0}$ in the above equations to get the threshold and probability of error.

(b) Note that $Y_2$ is simply $Y_1$ plus noise, and that noise is independent of $X$ and $Y_1$. Thus, $Y_2$, conditional on $Y_1$ and $X$ is simply $\mathcal{N}(Y_1, \sigma^2)$, which does not depend on $X$. Thus $Y_1$ is a sufficient statistic and $Y_2$ is irrelevant. Including $Y_2$ does not change the probability of error.

(c) It should have been clear intuitively that adding an additional observation that is only a noisy version of what has already been observed will not help in the decision, but knowledge of probability sharpens one’s intuition so that something like this becomes self evident without mathematical proof.

(d) The same argument as in (b) shows that $Y_2$, conditional on $Y_1$, is independent of $X$, and thus the decision rule and error probability do not change.

(e) If $Z_1$ is uniformly distributed between 0 and 1, then $Y_1$ lies between 5 and 6 for $X = 0$ and between 1 and 2 for $X = 1$. So the MAP rule leads to the decision $X = 1$ if $Y_1 \in [5, 6]$, and $X = 0$ if $Y \in [1, 2]$. There is no possibility of error in this case, so again $Y_2$ is irrelevant.

7. This problem will explore optimal classification based on the features selected via principal component analysis (PCA). Download the MNIST handwritten digit dataset from the course website [http://web.stanford.edu/class/ee278/homeworks/hw3-data.zip](http://web.stanford.edu/class/ee278/homeworks/hw3-data.zip). The folders `train0` and `train2` contain the same set of images we used in Homework 2. We will use these images to train a classifier that can distinguish between the digits “0” and “2”. We also have added two test sets (each with 500 images) of the handwritten digit “0” and of the digit “2” in the folders `test0` and `test2`.

a) As in Homework 2, consider each image as vector $X_i \in \mathbb{R}^{784}$. Combine all the training images (in folders `train0` and `train2`) and generate an estimator of the covariance matrix. Compute the first 20 eigenvectors $U_i (1 \leq i \leq 20)$ corresponding to the largest eigenvalues. Now project each training image $X_i$ onto the new set of basis vectors $U_i$. The result, denoted $\tilde{X}_i \in \mathbb{R}^{20}$, is a lower dimensional feature vector that we will use to represent the data.

b) Estimate the mean and covariance matrix of the training vectors $\tilde{X}_i$ corresponding to the digit “0” and the mean and covariance matrix corresponding to the digit “2”. Suppose that each digit-0 (resp. digit-2) image $\tilde{X}_i$ is independently drawn
from a jointly Gaussian distribution $P_0$ (resp. $P_2$). Propose a maximum likelihood detector that classifies a given image as a “0” or “2”, i.e. assuming an equal prior. (This is called Gaussian discriminative analysis in machine learning.)

c) Run your classifier on the test dataset in the folders test0 and test2. Report the empirical error rates.

Solution:

a) See the attached Matlab code.

b) For both train0 and train2, we use the following estimators to estimate the mean and covariance of the data

$$\hat{m}_0 = \frac{1}{n} \sum_{i \in \text{train0}} \tilde{X}_i, \quad \hat{K}_0 = \frac{1}{n} \sum_{i \in \text{train0}} (\tilde{X}_i - \hat{m}_0)(\tilde{X}_i - \hat{m}_0)^\top$$

$$\hat{m}_2 = \frac{1}{n} \sum_{i \in \text{train2}} \tilde{X}_i, \quad \hat{K}_2 = \frac{1}{n} \sum_{i \in \text{train2}} (\tilde{X}_i - \hat{m}_2)(\tilde{X}_i - \hat{m}_2)^\top$$

where $n = 4999$. For any new data $\tilde{X}_j$, we compute the log likelihood ratio

$$\text{LLR}\left(\tilde{X}_j\right) = \log \text{pdf}\left(\tilde{X}_j; \hat{m}_0, \hat{K}_0\right) - \log \text{pdf}\left(\tilde{X}_j; \hat{m}_2, \hat{K}_2\right),$$

where $\log \text{pdf}$ denotes the log likelihood of jointly Gaussian distribution

$$\log \text{pdf}(X; m, K) := -\frac{(X - m)^\top K^{-1}(X - m)}{2} - \frac{\log \det(K)}{2} - \frac{p}{2} \log(2\pi)$$  \hspace{1cm} (2)

where $p = 784$ is the dimension of the data. The maximum likelihood classification rule would be if $\text{LLR}\left(\tilde{X}_j\right) > 0$, then we declare $\tilde{X}_j$ represents digit “0”; and if $\text{LLR}\left(\tilde{X}_j\right) \leq 0$, then we declare $\tilde{X}_j$ represents digit “2”.

Note that we directly use the expression (2) to compute the log-likelihood for the sake of numerical stability.

c) The empirical error rate is 0.8%. The Matlab code is attached as follows.

```matlab
numImage = 4999;
N = 28;
X = zeros(N*N, numImage);
umPC = 20;

for i = 1: numImage
    filename0 = [ 'train0/', num2str(i,'%05d'), '.pgm' ];
    X(:,i) = reshape( imread( filename0 ), N^2, 1);
    filename2 = [ 'train2/', num2str(i,'%05d'), '.pgm' ];
    X(:,4999+i) = reshape( imread( filename2 ), N^2, 1);
```

```matlab
end
classifyX = classify(X, X);
n = size(X,2);
conf = confusionmatrix( classifyX, true labels );
error = 100*diag(conf,0) / n;
mean(error)
```
% remove the mean component
meanX = mean(X')';
centeredX = X - kron(meanX, ones(1, 2*numImage));
CovX = 1/2/numImage * centeredX * centeredX';

% eigen-decomposition / PCA
[U,S] = eigs(CovX, numPC, 'lm');
[S,I] = sort(diag(S),'descend');
U = U(:,I);

% project onto new basis
Train0 = U'* X(:, 1: numImage);
Train2 = U'* X(:, (numImage+1): (2*numImage));

% estimate the covariance / mean of digit-0 training images
meanTrain0 = mean(Train0,2);
CovTrain0 = Train0 - kron(meanTrain0, ones(1,numImage));
CovTrain0 = 1/numImage * CovTrain0 * CovTrain0';

% estimate the covariance / mean of digit-2 training images
meanTrain2 = mean(Train2,2);
CovTrain2 = Train2 - kron(meanTrain2, ones(1,numImage));
CovTrain2 = 1/numImage * CovTrain2 * CovTrain2';

% prediction
numTestImage = 500;
SuccessRate = 0;
for i = 0: (numTestImage-1)
    filename = [ 'test0/', num2str(i,'%05d'), '.pgm' ];
    Xt = U' * double( reshape( imread( filename ), N^2, 1) );
    LLR = logmvnpdf(Xt, meanTrain0, CovTrain0) - ... 
         logmvnpdf(Xt, meanTrain2, CovTrain2);
    if LLR > 0
        SuccessRate = SuccessRate + 1;
    end
end

for i = 0: (numTestImage-1)
    filename = [ 'test2/', num2str(i,'%05d'), '.pgm' ];
    Xt = U'* double( reshape( imread( filename ), N^2, 1) );
LLR = logmvnpdf(Xt, meanTrain0, CovTrain0) - ... 
logmvnpdf(Xt, meanTrain2, CovTrain2);

if LLR <= 0
    SuccessRate = SuccessRate + 1;
end
end

SuccessRate = SuccessRate / 2 / numTestImage;