Problem 1

logistic_reg.m: the in-sample error E_{in} is calculated by the formula $E_{in} = \frac{1}{N}\sum_{n=1}^{N} \ln(1 + e^{-y_n w^T x_n})$. The stopping criteria in logistic_reg.m is either that the maximum iterations is reached or the gradient becomes smaller than the threshold.

find_test_error.m: the script loop through each data point, calculate the prediction value and compare the prediction with the actual classification result to compute the classification error.

Experiments:

With maximum iteration set to 10000, 100000 and 1000000, and the tolerance set to 10^{-3}, the following results are obtained:

$\eta = 10^{-5}$ and $tolerance = 10^{-3}$ and $max\_its =$	10000	100000	1000000
E_{in}	0.5847	0.4937	0.4354
$E_{classification}(Test)$	0.3092	0.2237	0.1513
$E_{classification}(Train)$	0.3172	0.2069	0.1310

The actual result matches the expected algorithm behavior. The gradient descent algorithm moves towards minimizing E_{in} along the gradient. As a result, as more iterations are allowed, the weights are updated closer to the actual weights (assuming that the learning rate is correct) and therefore the classification errors for both training and testing data become less. However, the decrease in classification error is slower than the increase in the maximum iterations.

The MATLAB glmfit function returned a classification error of 0.1172 on the testing data, whereas on the training data, the classification error is 0.1776. The classification error is very close to the error 0.1310 we obtained from logistic_reg.m when maximum iteration is set to one million. However, glmfit function computes the weights really fast within seconds and while the gradient descent method is very slow and takes about ~15 minutes for one million maximum iterations.

With $max\_its = 10000$ and $tolerence = 10^{-6}$, running the z-scored data

Learning rate	iterations	E_{in}
1	228	0.4074
0.1	2318	0.4074
0.01	10000	0.4074
0.001	10000	0.4167
10^{-4}	10000	0.5237
10^{-5}	10000	0.6637

The table above is reasonable and matches the expectation. For small learning rate, each weight update only makes small change and therefore the weight makes a small move towards the correct weight per iteration. Therefore for the weight to approach the actual weight, we need a large number of iterations to compensate for the small learning rate. With the limitation of at most 10000 iterations, E_{in} becomes bigger as the learning rate becomes really samll because after 10000 iterations of learning, we still learned little due to the small learning rate.

If we have a bigger learning rate, the iterations needed is less because we quickly learned the correct result and further updates are smaller than the tolerance. However, we may never be able to reach the actual minimum if the learning rate is too large.

Problem 2 – LFD problem 3.19

(a) The transformation can become problematic if the data set size N becomes large.

In short, the data is transformed into the space whose dimension is totally dependent on the size of the data set. For a data size of size N, we have

$$dimension\big(\Phi(x)\big) = N$$

That is, each data point represents one dimension in the transformed space and there are no duplicate transformed data inputs. Therefore, the maximum dichotomies can be $m_H(N) = 2^N$.

As a result, the VC dimension is N and when the sample size is large, the VC dimension is also large.

Therefore the transformation is problematic when sample size is large.

(b) $\phi_{\mathrm{n}}(x)$ is a valid radial basis function and it calculates the distance between x and x_n in the transformed space. RBF is a wildly used kernel function.

(c) Similarly, $\phi_{\mathrm{i,j}}(x)$ is also a valid radial basis function.

Problem 3 – LDF exercise 4.5

(a) It is very clear that $\sum_{q=0}^{Q} w_q^2 = w^T w$. Therefore, the constraint $\sum_{q=0}^{Q} w_q^2 \leq C$ becomes $w^T w \leq C$.

With the Tikhonov regularization constraint $w^T \Gamma^T \Gamma w \leq C$, it is clear that Γ is an identity matrix $\Gamma = I$.

(b) The constraint $\left(\sum_{q=0}^{Q} w_q\right)^2 \leq C \leftrightarrow \left(\sum_{q=0}^{Q} w_q\right)\left(\sum_{q=0}^{Q} w_q\right) \leq C$ implies that, in Tikhonov regularization constraint $w^T \Gamma^T \Gamma w \leq C$, $w^T \Gamma^T = \sum_{q=0}^{Q} w_q$. That is, Γ is a $1 \times n$ matrix of ones $\Gamma = (1\ 1 \ldots 1)$.

Problem 4 – LDF problem 4.8

Compute the gradient of the augmented error $E_{aug}(w) = E_{in}(w) + \lambda w^T w$, we have

$$\nabla E_{aug}(w) = \nabla E_{in}(w) + 2\lambda w$$

The update rule $w(t+1) \leftarrow w(t) - \eta \nabla E_{aug}\big(w(t)\big)$ therefore can be rewritten as:

$$w(t+1) \leftarrow w(t) - \eta \left(\nabla E_{in}\big(w(t)\big) + 2\lambda w(t)\right) = w(t+1) \leftarrow (1 - 2\eta\lambda) w(t) - \nabla E_{in}\big(w(t)\big)$$