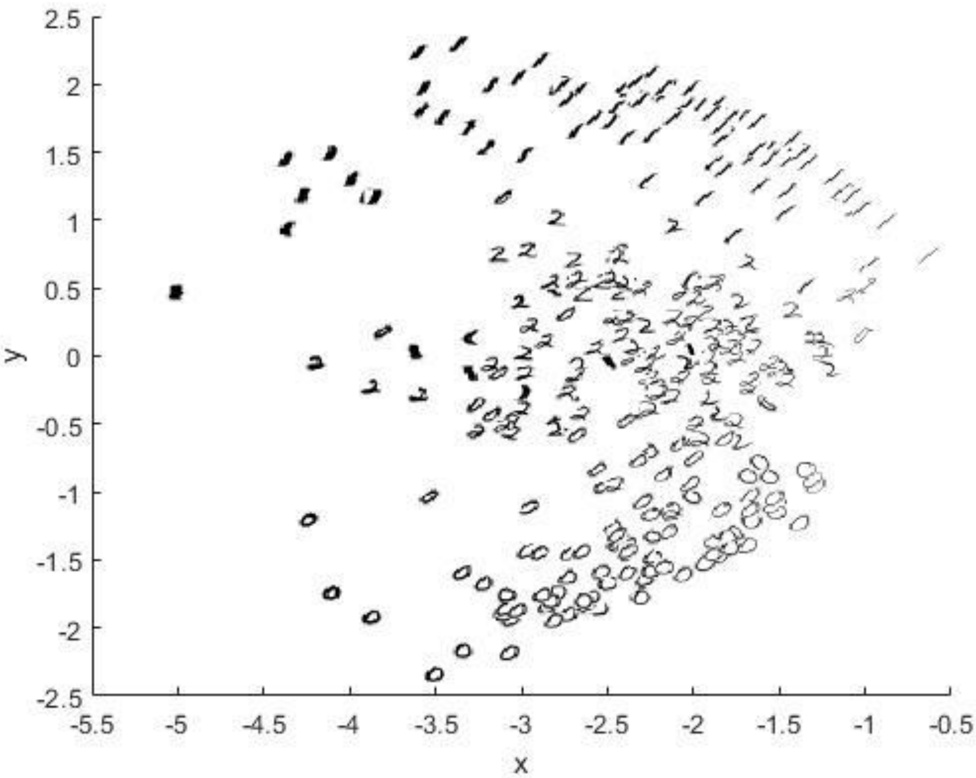
a

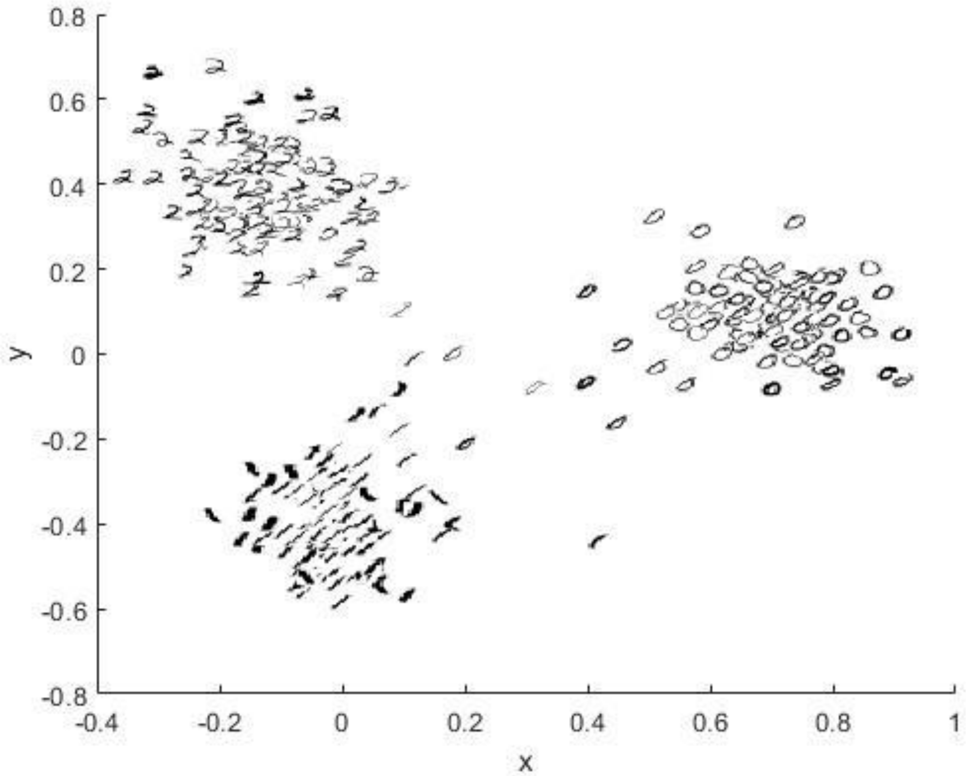
```
%(a)
[U, S, ~] = svd(X);
Y_pca = U(:, 1:2)' * X;
plotimages(reshape(X,8,8,300), Y_pca, 0.1, 1);

%(b)
```



b

```
%(b)
X1 = X(:, 1:100);
X2 = X(:, 101:200);
X3 = X(:, 201:300);
Sw = cov(X1') + cov(X2') + cov(X3');
[d, n] = size(X);
St = cov(X');
Sb = St - Sw;
W = Sw \ Sb;
[V, D] = eig(W);
Y_fda = V(:, 1:2)' * X;
plotimages(reshape(X,8,8,300), Y_fda, 0.03, 1);
```



c

```
%(c)
mu0 = mean(Y_pca(:, 1:100), 2);
mu1 = mean(Y_pca(:, 101:200), 2);
mu2 = mean(Y_pca(:, 201:300), 2);
sig0 = 1 / (100 - 2) * (Y_pca(:, 1:100) - mu0) * (Y_pca(:, 1:100) - mu0)';
sig1 = 1 / (100 - 2) * (Y_pca(:, 101:200) - mu1) * (Y_pca(:, 101:200) - mu1)';
sig2 = 1 / (100 - 2) * (Y_pca(:, 201:300) - mu2) * (Y_pca(:, 201:300) - mu2)';
sig = 1 / 3 * (sig0 + sig1 + sig2);
findboundary(mu0, mu1, sig);
findboundary(mu0, mu2, sig);
findboundary(mu1, mu2, sig);

Y_pca_q = [Y_pca; Y_pca(1, :).^2; Y_pca(2, :).^2];
mu0_q = mean(Y_pca_q(:, 1:100), 2);
mu1_q = mean(Y_pca_q(:, 101:200), 2);
mu2_q = mean(Y_pca_q(:, 201:300), 2);
sig0_q = 1 / (100 - 4) * (Y_pca_q(:, 1:100) - mu0_q) * (Y_pca_q(:, 1:100) - mu0_q)';
sig1_q = 1 / (100 - 4) * (Y_pca_q(:, 101:200) - mu1_q) * (Y_pca_q(:, 101:200) - mu1_q)';
sig2_q = 1 / (100 - 4) * (Y_pca_q(:, 201:300) - mu2_q) * (Y_pca_q(:, 201:300) - mu2_q)';
sig_q = 1 / 3 * (sig0_q + sig1_q + sig2_q);
findboundary(mu0_q, mu1_q, sig_q);
findboundary(mu0_q, mu2_q, sig_q);
findboundary(mu1_q, mu2_q, sig_q);
```

```
function A = findboundary(mu0, mu1, sig)
    temp = mu1' / sig - mu0' / sig;
    [~, n] = size(temp);
    A = reshape([temp, 1/2 * (mu0' / sig * mu0 - mu1' / sig * mu1)], 1, n + 1);
end
```

The analytic forms to LDA is

$$-0.4289x + 8.6491y - 1.3464 = 0$$

$$-0.0084x + 4.2779y + 2.6426 = 0$$

$$0.4205x - 4.3712y + 3.9890 = 0$$

The analytic forms to QDA is

$$6.9541x + 8.6151y + 1.4132\mathrm{x}^2 + 0.2228\mathrm{y}^2 + 6.8859 = 0$$

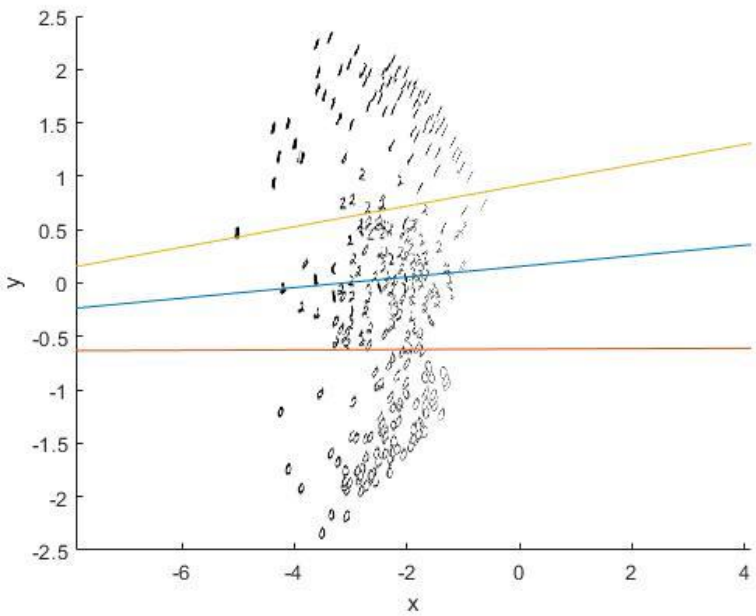
$$-3.6052x + 4.3923y - 0.5827\mathrm{x}^2 - 1.9371\mathrm{y}^2 - 0.2325 = 0$$

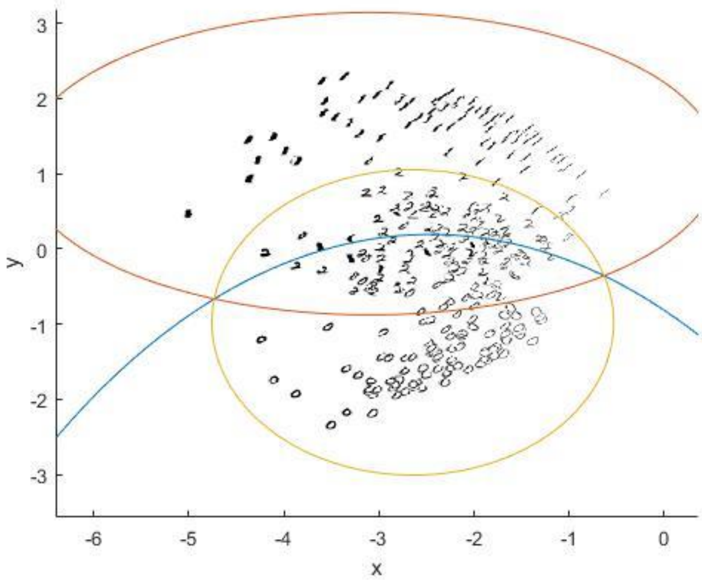
$$-10.5593x - 4.2228y - 1.9959\mathrm{x}^2 - 2.1600\mathrm{y}^2 - 7.1184 = 0$$

d

```
%d
plotimages(reshape(X,8,8,300), Y_pca, 0.1, 1);
hold on;
fimplicit(@(x, y) (mu1' - mu0') / sig * [x;y] + 1/2 * (mu0' / sig * mu0 - mu1' / sig * mu1));
fimplicit(@(x, y) (mu2' - mu0') / sig * [x;y] + 1/2 * (mu0' / sig * mu0 - mu2' / sig * mu2));
fimplicit(@(x, y) (mu1' - mu2') / sig * [x;y] + 1/2 * (mu2' / sig * mu2 - mu1' / sig * mu1));


plotimages(reshape(X,8,8,300), Y_pca, 0.1, 1);
hold on;
fimplicit(@(x, y) (mu1_q' - mu0_q') / sig_q * [x;y;x^2;y^2] + 1/2 * (mu0_q' / sig_q * mu0_q - mu1_q' / sig_q * mu1_q));
fimplicit(@(x, y) (mu2_q' - mu0_q') / sig_q * [x;y;x^2;y^2] + 1/2 * (mu0_q' / sig_q * mu0_q - mu2_q' / sig_q * mu2_q));
fimplicit(@(x, y) (mu1_q' - mu2_q') / sig_q * [x;y;x^2;y^2] + 1/2 * (mu2_q' / sig_q * mu2_q - mu1_q' / sig_q * mu1_q));
```





e

```
function num = lda(X, mu)
    [~, n] = size(mu);
    min = (X - mu(:,1))' * (X - mu(:,1));
    num = 0;
    for i = 2:n
        temp = (X - mu(:,i))' * (X - mu(:,i));
        if min > temp
            min = temp;
            num = i - 1;
        end
    end
end
function e = error_rate(X)
    e = 0;
    mu0 = mean(X(:, 1:100), 2);
    mu1 = mean(X(:, 101:200), 2);
    mu2 = mean(X(:, 201:300), 2);
    mu = [mu0, mu1, mu2];
    for i = 1:300
        if 1 <= i && i <= 100 && lda(X(:, i), mu) ~= 0
            e = e + 1;
        elseif 101 <= i && i <= 200 && lda(X(:, i), mu) ~= 1
            e = e + 1;
        elseif 201 <= i && i <= 300 && lda(X(:, i), mu) ~= 2
            e = e + 1;
        end
    end
    e = e / 300;
end
```

The error rate is 0.0733

f

Yes. We can augment every x by adding $\mathrm{x_i}^2$ to the vector and follow the exact same process in part (e) to perform QDA. Since the QDA in part (c) has different dimension and $\delta(\mathrm{x})$. It is not identical to the QDA in c.