

## HW14 M9

Answer (for reference only)

1.

If we run the code below directly without modifying the package provided by professor Jang:

```
DS = prData('abalone');
DS.input=DS.input(5:6, :);
[qcPrm, logProb, recogRate, hitIndex]=qcTrain(DS);
DS.hitIndex = hitIndex;
qcPlot(DS, qcPrm, 'decBoundary');
axis square
```

It would return these errors:

```
Error using -
Matrix dimensions must agree.
Error in classifierEval (line 78)
        dataMinusMu = DS.input-cPrm.class(i).mu*ones(1, dataNum);
```

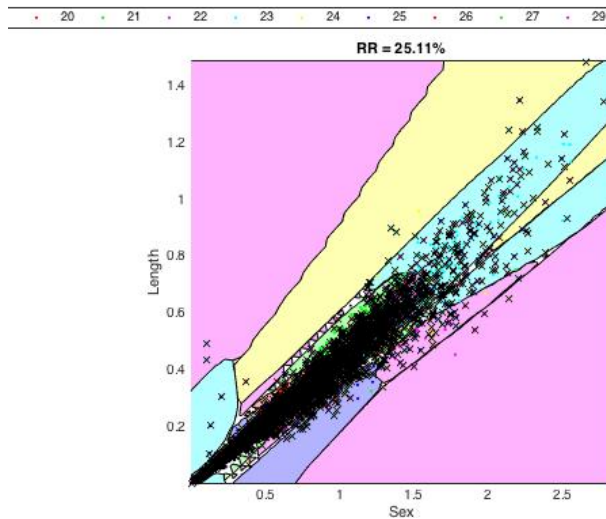
I trace the code from the error message and find that in the gaussianMle.m function, line 40:

```
if size(data, 2)==1, data=data'; end
```

It transposed the matrix when the data size of the class is only one, so the dimension would be incompatible.

After I commented the line, it can run the result successfully.

The result figure:



2.

- Why the classifier is named "quadratic"?

The resulting classifier is often called the quadratic classifier (QC) since the decision boundary between any two classes is a quadratic function in the feature space.

- How to train?

Identify the Gaussian PDF of each class via MLE

- How to test?

Assign a sample point to the class C by taking the class prior into consideration:

$$\hat{C} = \underset{C}{\operatorname{argmax}} \Pr(C) * Pdf_C(\mathbf{x})$$

- Strength?

> Efficient computations when the dimension d is small  
> Efficient way to compute leave-one-out cross validation

- Weakness?

> The covariance matrix (d by d) is big when the dimension d is median large  
> The inverse of the covariance matrix may not exist  
> Cannot handle bi-modal data