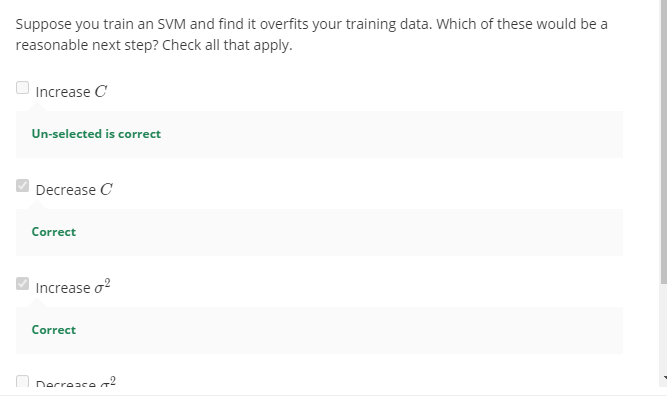
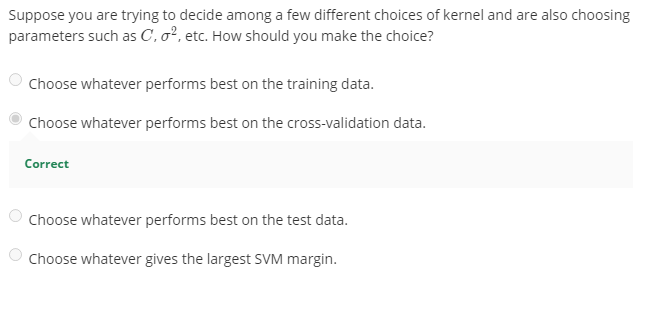
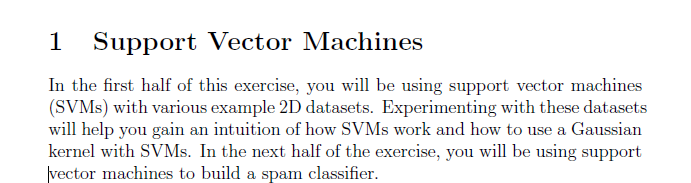
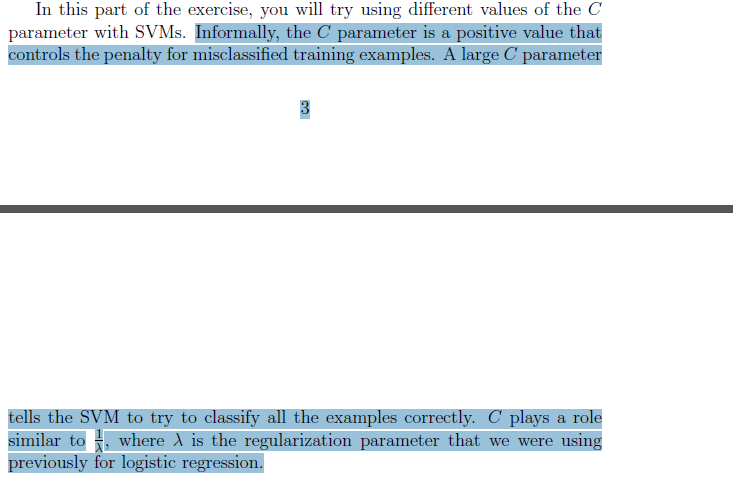
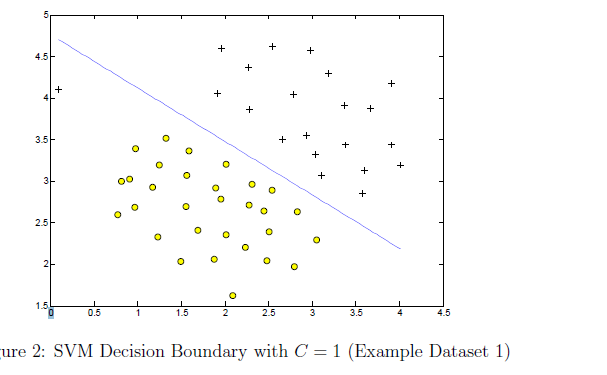
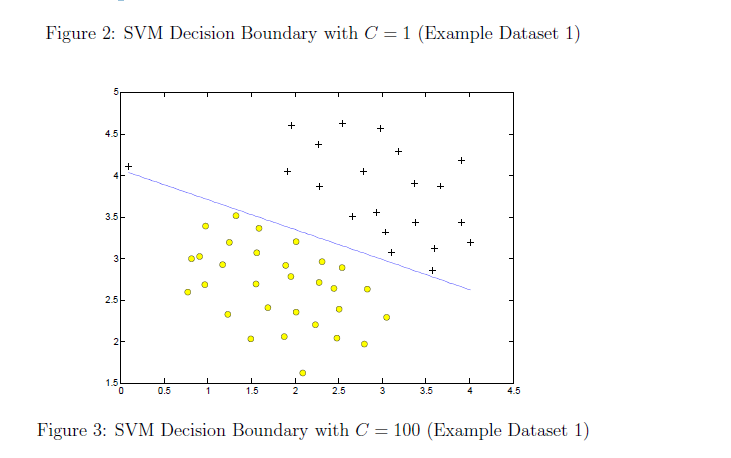
1. 
2. 

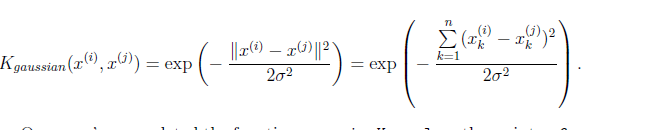




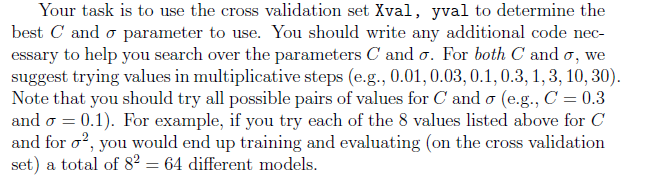


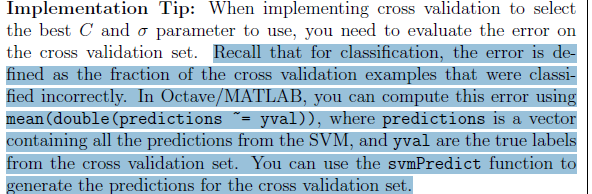


1. **gaussianKernel.m :**



### dataset3Params.m :





**Code :**

Two loops is there

**C\_test = C\_list(i);**

**sigma\_test = sigma\_list(j);**

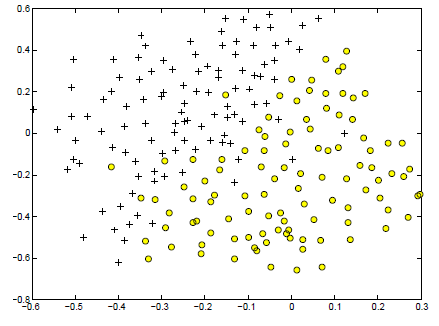
**model = svmTrain(X, y, C\_test, @(x1, x2) gaussianKernel(x1, x2, sigma\_test));(***svm Train is already built In function X whole datset( x1,lables, C and sigma to train*  ***)***

**predictions = svmPredict(model, Xval);(***train model and X to predict the classification* **)**

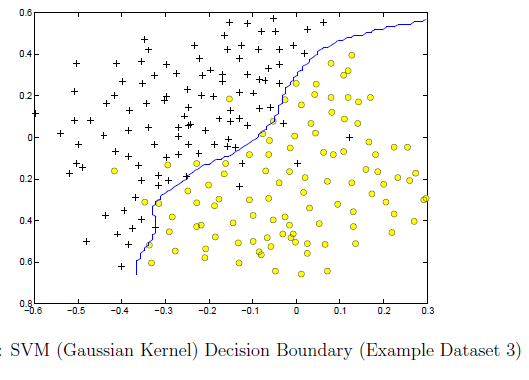
**prediction\_error(i,j) = mean(double(predictions ~= yval**

Then use the logic to get C and sigma which gives minimum error

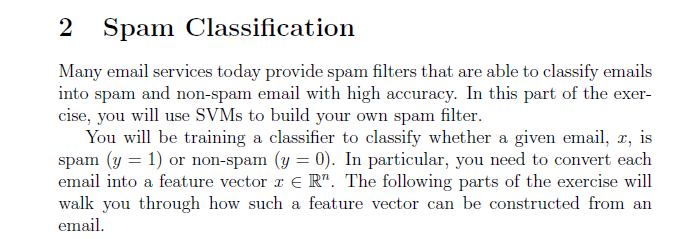
**Input(Takes around 1.5 minutes ):**



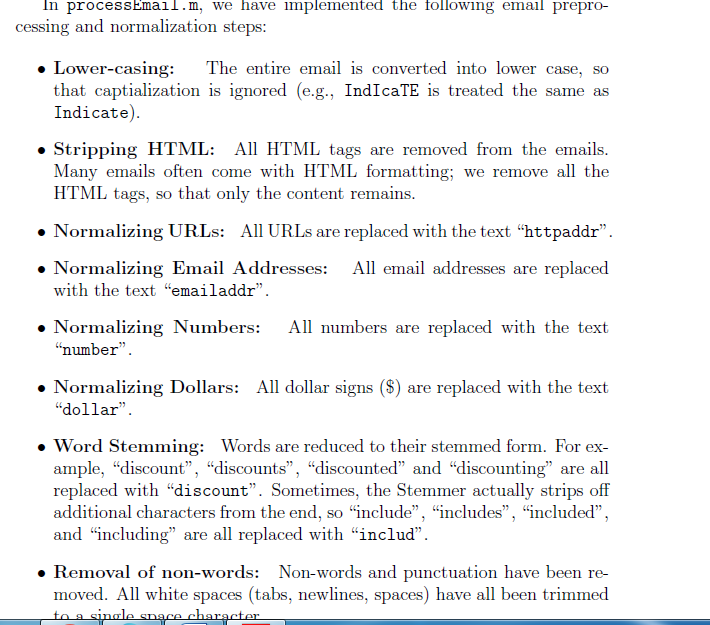
**Output (Non Linear Boundary that Gausian Classifier made with help of C and sigma value ) :**

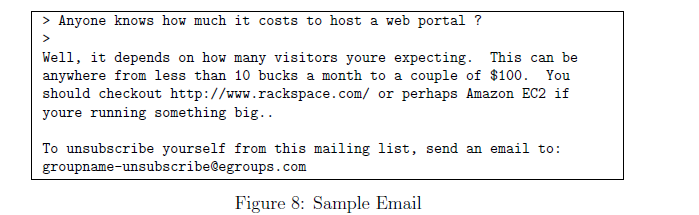


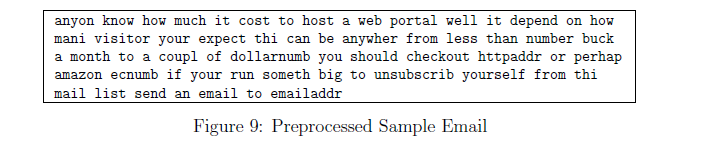
**Part 2 : for the Exercise**

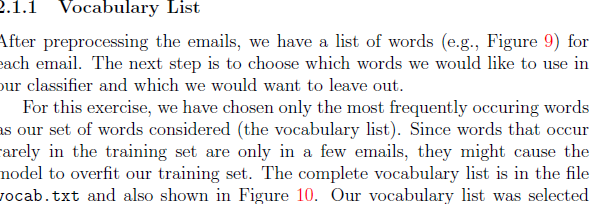


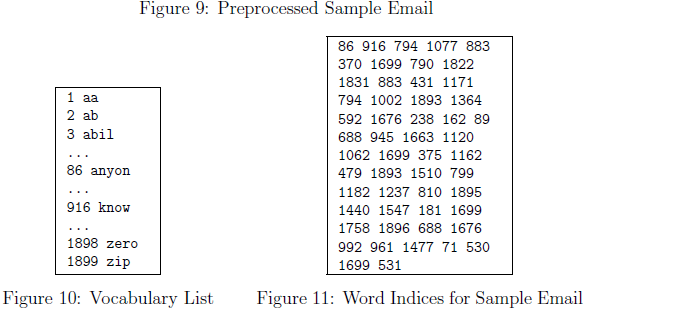
**Pre Processing a Email**











### processEmail.m :

### 

### emailFeatures.m :

### 

### 

### 

### No need to code for this

### 

### Code implemented

### 

### 

### Actual Final; Prediction whether it’s spam or not

### 

### Code :

### 

### 

**Data Prepocessing of Email**

**Step 1 : Guassian kernel formula**

**Step 2 : Best C and Sigma Values**

**Step 3 : train the SVM using svm Model(Where we pass the C,sigma, Gaussian Kernel value )**

**Step : svmPredict (1 or 0)**