ex6

第一部分 支持向量机

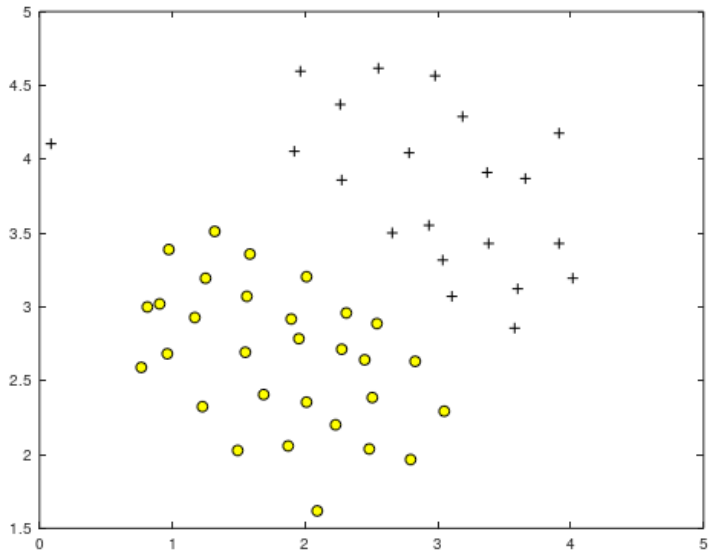
第一部分练习对不同样本的2D数据集使用支持向量机，对支持向量机的运行方式以及如何使用核函数拥有直观的认识。第二部分练习使用支持向量机生成一个垃圾邮件分类器。

1.1 样本数据集1

（1）数据显示

从一个能被线性边界分开的二维样本数据集开始，ex6.m显示训练数据，注意有一个正例为异常值，我们将会了解异常值（离群值）是如何影响支持向量机的决策边界的。

```
load('ex6data1.mat');
plotData(X, y);
```



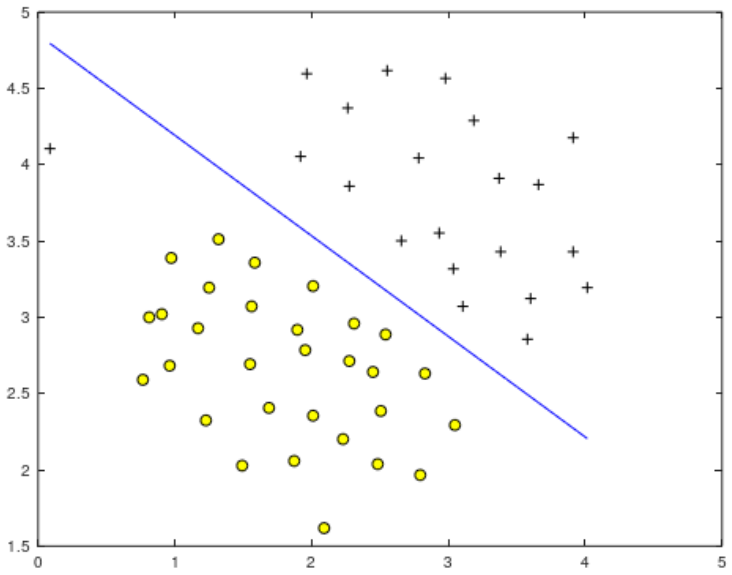
在本练习中尝试对不同的支持向量机使用不同的C值。不正式地说，C是控制对错误分类的训练样本的惩罚的正值，较大的C值会告诉支持向量机对所有的训练样本准确地分类，扮演类似于1/λ的角色，λ则是我们以前在逻辑回归中使用过的正则化参数。

（2）训练线性支持向量机

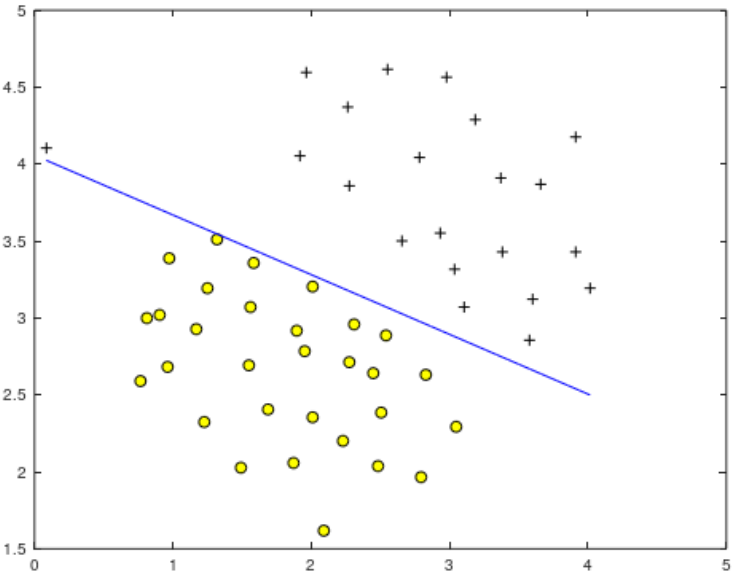
训练线性支持向量机并显示出决策边界

```
load('ex6data1.mat');
```

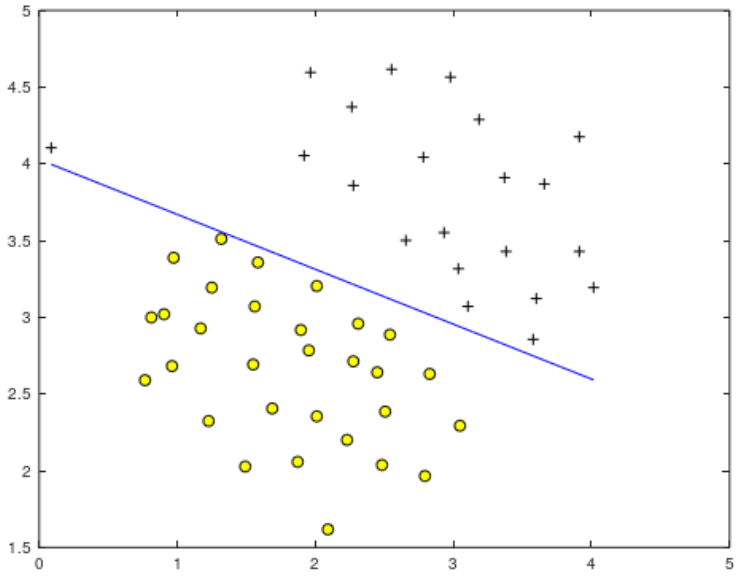
```
fprintf('\nTraining Linear SVM ...\n')
C = 1;
model = svmTrain(X, y, C, @linearKernel, 1e-3, 20);
visualizeBoundaryLinear(X, y, model);
```



尝试使用不同的C值，比如100，所有数据均准确分类，但是并不自然



尝试使用不同的C值，比如1000



主要使用了svmTrain.m文件中的SVM software训练支持向量机，注意在C仅仅为1的时候SVM将异常值分类错误。

注：大多数SVM software（包括svmTrain.m）自动添加额外特征值x0=1，自动学习截距参数θ0，当我们传数据到SVM software中去的时候，不需要手动添加x0=1。

1.2 高斯核函数

使用支持向量机进行非线性分类，尤其是使用带高斯核函数的支持向量机对无法进行线性划分的数据集进行分类。

1.2.1 高斯核函数

为了用支持向量机找到非线性的决策边界，我们首先需要实现一个高斯核函数。可以将高斯核函数想象成一个用来衡量一对样本（Xi，Xj）之间“距离”的相似度函数，高斯函数也有一个带宽参数σ，决定了当两个样本相距越来越远的时候，相似度的下降速度。

完成gaussianKernel.m文件，用来计算两个样本（Xi，Xj）之间的高斯核，高斯核函数被如下定义：

$$K_{gaussian}(x^{(i)}, x^{(j)}) = \exp\left(-\frac{\|x^{(i)} - x^{(j)}\|^2}{2\sigma^2}\right) = \exp\left(-\frac{\sum_{k=1}^{n}(x_k^{(i)} - x_k^{(j)})^2}{2\sigma^2}\right)$$

完成该脚本之后，运行ex6.m文件对两个样本进行高斯核函数测试

代码：

```
sim=exp(sum((x1-x2) .^2)/(-2*sigma.^2))
```

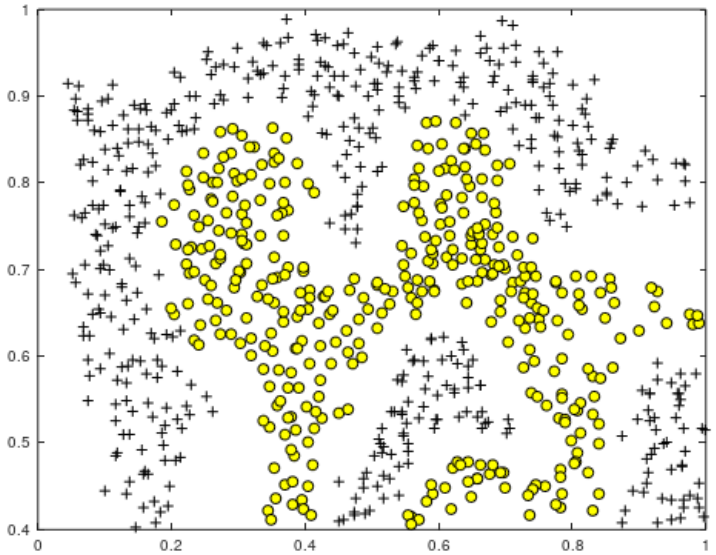
运行结果：

```
Evaluating the Gaussian Kernel ...
sim =  0.32465
Gaussian Kernel between x1 = [1; 2; 1], x2 = [0; 4; -1], sigma = 2.000000 :
        0.324652
(for sigma = 2, this value should be about 0.324652)
```

第一次提交：

Part Name	Score	Feedback
Gaussian Kernel	25 / 25	Nice work!
Parameters (C, sigma) for Dataset 3	0 / 25	
Email Preprocessing	0 / 25	
Email Feature Extraction	0 / 25	
	25 / 100	

1.2.2 样本数据集2



load('ex6data2.mat');

plotData(X, y);

加载并显示数据集2，从图中我们可以看到没有线性决策边界能够将训练集中的正例和负例样本区分开来，但是通过SVM的高斯核函数，我们能够学习出一个非线性决策边界来较好地区分数据集，完成高斯核函数后，ex6.m文件将会使用它训练支持向量机。

ex6.m代码:

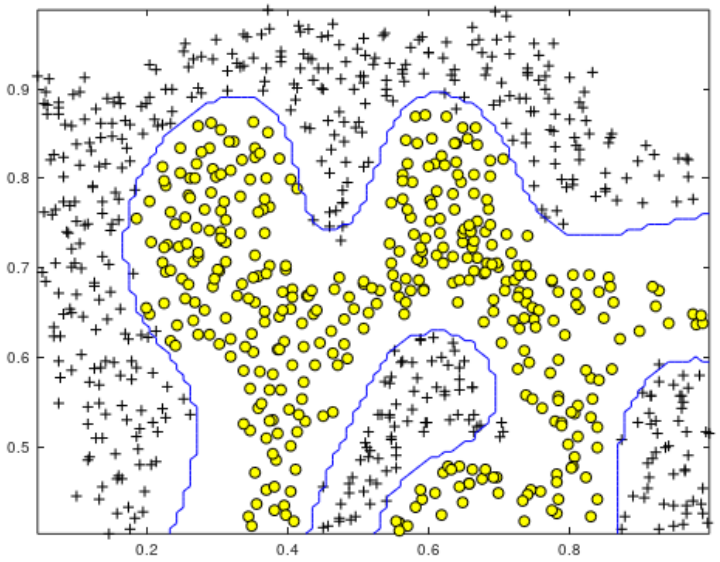
load('ex6data2.mat');

C = 1; sigma = 0.1;

model= svmTrain(X, y, C, @(x1, x2) gaussianKernel(x1, x2, sigma));

visualizeBoundary(X, y, model);

运行结果：



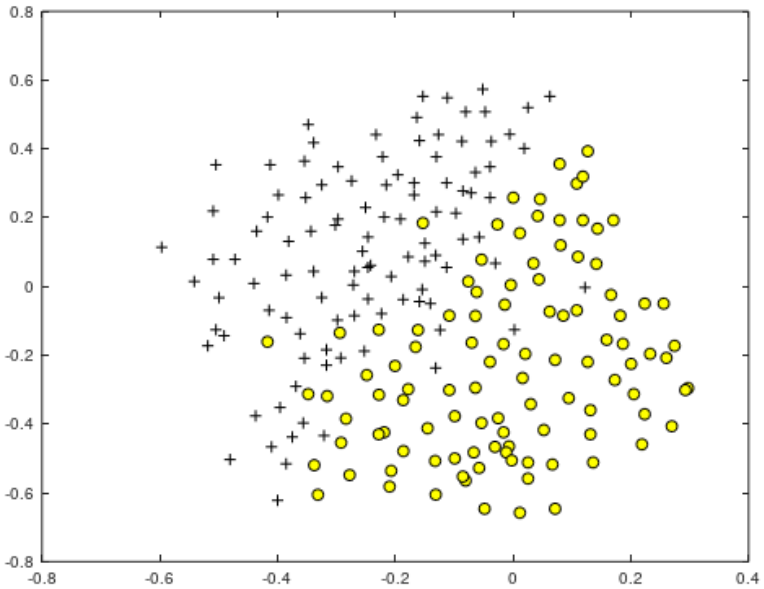
该图展示了支持向量机通过高斯核函数获取到的决策边界，其能够正确地区分大多数正例和负例样本，很好地拟合数据集的轮廓。

1.2.3 样本数据集3

本练习中将会学习更多使用带有高斯核函数的支持向量机的实战技巧，运行ex6.m将会加载并显示第三个数据集，我们将会对其使用带高斯核函数的支持向量机

load('ex6data3.mat');

plotData(X, y);



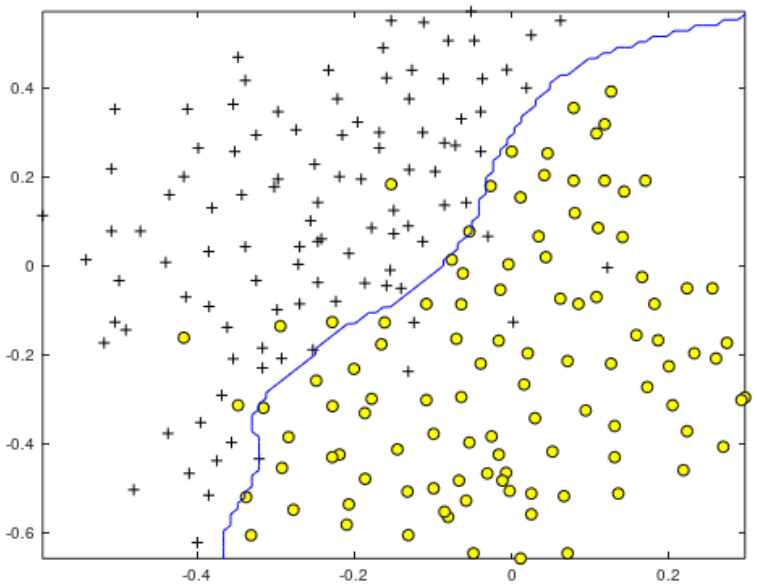
这里我们拥有变量X，y，Xval以及yval，ex6.m文件通过datasetParams.m文件中的参数使用训练集（X，y）训练支持向量机，我们的任务是通过交叉验证机集决定超参数C和σ的最优值，这两个超参数的尝试值均建议设置为：（0.01,0.03,0.1,0.3,1,3,10,30），注意每一个C和σ对均需要进行测试生成一种模型，也就是共产生64个模型。在决定C和σ的值后，调整dataset3Params.m文件中的代码，将找到的最优C和σ填入，此时的SVM将会产生一条决策边界。

注意：为了选出最优的C和σ，我们需要计算模型在交叉验证集上的误差，使用mean(double(predictions ~= yval))进行计算，predictions是预测向量，yval是真实判断，预测向量Predictions可以使用svmPredict.m文件生成。

代码：

```
C_values=[0.01,0.03,0.1,0.3,1,3,10,30];
sigma_values=[0.01,0.03,0.1,0.3,1,3,10,30];
for i=1:length(C_values)
  for j=1:length(sigma_values)
        model= svmTrain(X, y, C_values(i), @(x1, x2) gaussianKernel(x1, x2, sigma_values(j)));
    predictions=svmPredict(model,Xval);
    predictions_error(i,j)=mean(double(predictions~=yval));
  endfor
endfor
miniest=min(min(predictions_error));
[i,j]=find(predictions_error==miniest);
C=C_values(i);
sigma=sigma_values(j);
```

运行结果：



第二次提交：

```
                             Part Name |     Score | Feedback
                             --------- |     ----- | --------
                         Gaussian Kernel |  25 /  25 | Nice work!
Parameters (C, sigma) for Dataset 3 |  25 /  25 | Nice work!
                       Email Preprocessing |   0 /  25 |
                  Email Feature Extraction |   0 /  25 |
                                           --------------------------------
                                           |  50 / 100 |
```

第二部分 垃圾邮件分类器

在本节中将使用SVM构建属于自己的垃圾邮件分类系统，需要将邮件转换成一个特征向量，接下来将会学习如何从一个邮件中构建一个特征向量。使用ex6_spam.m文件，数据集来自 SpamAssassin Public Corpus。我们将只使用邮件主体而不使用邮件标题。

2.1 对邮件进行预处理

在开始机器学习任务之前通常会对数据集中的 样本进行观察，下图表明该样本邮件包含一个URL、一个邮件地址（最后）、数字以及美金数量，尽管很多邮件也会包含相似类型的字符，特定的字符（比如特定的URL以及特定的邮箱地址）在几乎每一份邮件中都是不一样的，因此预处理邮件的常见做法之一是对这些值进行标准化处理，这样所有的URL是一样的表示，所有的数字也是一样的表示。举例来说，我们将所有的URL全部使用特定的字符串“httpaddr"来进行表示，这样可以让垃圾邮件分类器根据URL是否存在进行分类决策，而不是根据一个特定的URL是否存在来进行分类决策。这是一种提升垃圾邮件分类器性能的典型方法，因为来及邮件发送者经常随机发送URL，在新邮件中再次看到以前见过的URL的概率很小。

```
> Anyone knows how much it costs to host a web portal ?
>
Well, it depends on how many visitors youre expecting.  This can be
anywhere from less than 10 bucks a month to a couple of $100.  You
should checkout http://www.rackspace.com/ or perhaps Amazon EC2 if
youre running something big..

To unsubscribe yourself from this mailing list, send an email to:
groupname-unsubscribe@egroups.com
```

在processEmail.m文件中，我们实现以下对邮件进行预处理和标准化的步骤：

（1）所有字母全部小写

（2）剥离Html标签，只保留邮件内容

（3）标准化URLs，全部用httpaddr替代

（4）标准化邮箱地址，全部用emailaddr替代

（5）标准化数字，全部用number替代

（6）标准化美元数字，全部使用dollar替代

（7）保留词干，单词减小为保留词干的形式，比如“discount”，“discounts”，“discounted” a和“discounting” 全部用"discount“来代替，通常从单词的末尾开始剥离

（8）删除非单词以及标点符号，所有白空格 (tabs, newlines, spaces)修改为空格符号

预处理的结果如下图所示，尽管预处理出现了单词碎片和非单词，却使得特征提取变得容易。

```
anyon know how much it cost to host a web portal well it depend on how
mani visitor your expect thi can be anywher from less than number buck
a month to a coupl of dollarnumb you should checkout httpaddr or perhap
amazon ecnumb if your run someth big to unsubscrib yourself from thi
mail list send an email to emailaddr
```

2.1.1 词汇表

下一步是挑选我们想要在分类器中使用的单词，以及我们想舍弃的单词。在本次练习中，我们只选择出现频率最高的作为词汇表。由于在训练集中很少出现的单词只出现在几封邮件中，它们可能会导致我们过拟合训练集，完整的词汇表在vocab.txt文件中。我们词汇表是在垃圾邮件语料库中的至少出现100次的单词，大概有1899个。通常在实践操作过程中，词汇表会包含10000-50000个单词。

在拥有词汇表之后，我们现在将处理过的邮件中的每一个单词映射到一个词汇索引表中，其元素均是词汇表中单词的索引。特别地，在样本邮件中，anyone这个单词被首先归一化为anyon，然后映射到词汇表中的第86号索引。

你现在的任务是完成processEmail.m文件中的代码已执行上述映射的操作。在代码中，str是一个string类型的变量，表示预处理邮件中的单个单词。在词汇表vocabList中查阅这个单词看看它是否存在于词汇表中。如果确实存在，将该单词的索引添加到word_indices变量中去，如果不存在于词汇表中，则跳过这个单词。

完成processEmail.m文件之后，运行ex6_spam在样本邮件上。

注意：在Octave中，使用strcmp函数进行string类型值的比较操作，例如如果str1与str2相同，则strcmp(str1,str2)返回1，vocabList是一个包含了词汇表中单词的元组，Ocatve中的元组就是一个向量，只是其元素也能是字符串，而真正的矩阵或者是向量不能包含字符串，我们使用大括号而不是方括号进行索引。使用length(vocabList)同样能获得词汇表中单词的数量。

代码：

```
for j=1:length(vocabList)
  if strcmp(str,vocabList{j})==1
    word_indices = [word_indices ; j];
  endif
endfor
```

运行结果：

```
Word Indices:
 86 916 794 1077 883 370 1699 790 1822 1831 883 431 1171 794 1002 1893 1364 592 1676 238 162 89 688 945 1663 1120 1062 1699 375 1162 479 1893 1510 799 1182 1237 810 1895 1440 1547 181 1699
1758 1896 688 1676 992 961 1477 71 530 1699 531
```

第三次提交：

```
                  Part Name |     Score | Feedback
                  --------- |     ----- | --------
                Gaussian Kernel |  25 /  25 | Nice work!
Parameters (C, sigma) for Dataset 3 |  25 /  25 | Nice work!
             Email Preprocessing |  25 /  25 | Nice work!
           Email Feature Extraction |   0 /  25 |
                                   --------------------------------
                                             |  75 / 100 |
```

2.2 从邮件中抓取特征

通过将每个样本转换成n维向量来实现特征抓取，n表示词汇表中单词的数量，特征x(i)属于{0,1}，判断字典中的第i个单词是否出现在邮件中，对于一封邮件，其特征应表示为：

$$x = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^n$$

完成emailFeatures.m文件，来为邮件生成一个特征向量，运行ex6_spam.m文件，样本邮件的特征向量是1899维的，拥有45个非零元素。

代码：

```
for i=1:length(word_indices)
  x(word_indices(i))=1;
endfor
```

运行结果：

```
Length of feature vector: 1899
Number of non-zero entries: 45
```

第四次提交：

```
                  Part Name |     Score | Feedback
                  --------- |     ----- | --------
                Gaussian Kernel |  25 /  25 | Nice work!
Parameters (C, sigma) for Dataset 3 |  25 /  25 | Nice work!
             Email Preprocessing |  25 /  25 | Nice work!
           Email Feature Extraction |  25 /  25 | Nice work!
                                   --------------------------------
                                             | 100 / 100 |
```

2.3 为垃圾邮件分类训练支持向量机

完成特征抓取函数后，运行ex6_spam.m将会加载一个与处理过的训练数据集，用来训练支持向量机分类器。 spamTrain.mat包含4000个垃圾邮件/非垃圾邮件训练样本，spamTest.mat包含1000个测试样本。每个初始邮件均运行了processEmail、emailFeatures函数并被转换成1899维的x(i)特征向量。

数据集加载完成后， ex6 spam.m将会训练SVM来对垃圾邮件和非垃圾邮件进行分类，训练完成后可以得到一个训练集准确率99.8%、测试集准确率98.5%的SVM支持向量

机。

```
Training ..........................................................
..........................................................
..........................................................
.................................................... Done!

Training Accuracy: 99.825000

Evaluating the trained Linear SVM on a test set ...
Test Accuracy: 98.800000
```

2.4垃圾邮件最容易被检测出来的特征

为了更好地理解垃圾邮件分类器的工作方式，我们检查那些最容易被检测出是垃圾邮件的特征，运行ex6_spam.m文件，找到相关词汇，如果邮件中拥有以下词汇，则会被分类为垃圾邮件。

```
Top predictors of spam:
 our              (0.499719)
 click            (0.463747)
 remov            (0.421564)
 guarante         (0.390800)
 visit            (0.373025)
 basenumb         (0.347244)
 dollar           (0.327315)
 will             (0.271427)
 price            (0.262935)
 pleas            (0.260318)
 most             (0.258577)
 nbsp             (0.258061)
 lo               (0.257732)
 ga               (0.244070)
 al               (0.241040)
```

2.5 可选练习：使用自己的邮件

使用spamSample1.txt样本进行分类：

```
do you want to make dollarnumb or more per week if you ar a motiv and qualifi
individu i will person demonstr to you a system that will make you dollarnumb
number per week or more thi is not mlm call our number hour pre record number
to get the detail number number number i need peopl who want to make seriou
monei make the call and get the fact invest number minut in yourself now
number number number look forward to your call and i will introduc you to
peopl like yourself who ar current make dollarnumb number plu per week number
number number numberljgvnumb numberleannumberlrmsnumb
numberwxhonumberqiytnumb numberrjuvnumberhqcfnumb numbereidbnumberdmtvlnumb

========================

Processed spamSample1.txt

Spam Classification: 1
(1 indicates spam, 0 indicates not spam)
```

使用spamSample2.txt样本进行分类：

```
best bui viagra gener onlin viagra numbermg x number pill dollarnumb free
pill reorder discount top sell number qualiti satisfact guarante we accept
visa master e check payment number satisfi custom httpaddr

========================

Processed spamSample2.txt

Spam Classification: 1
(1 indicates spam, 0 indicates not spam)
```

2.6 可选练习：创建自己的数据集

使用来自SpamAssassin Public Corpus（网站http://spamassassin.apache.org/old/publiccorpus/）上的原始文件，通过processEmail以及emailFeatures函数抓取特征向量后生成数据集，随机分成训练集、交叉验证集和测试集。建议通过选择出现在数据集中的高频词汇生成词汇表，添加你认为有用的

额外特征。最后建议使用高度优化过的SVM工具盒如LIBSVM（网址https://www.csie.ntu.edu.tw/~cjlin/libsvm/）。