实验报告-ex6

编程软件：MATLAB，notepad++

编程环境：Windows10

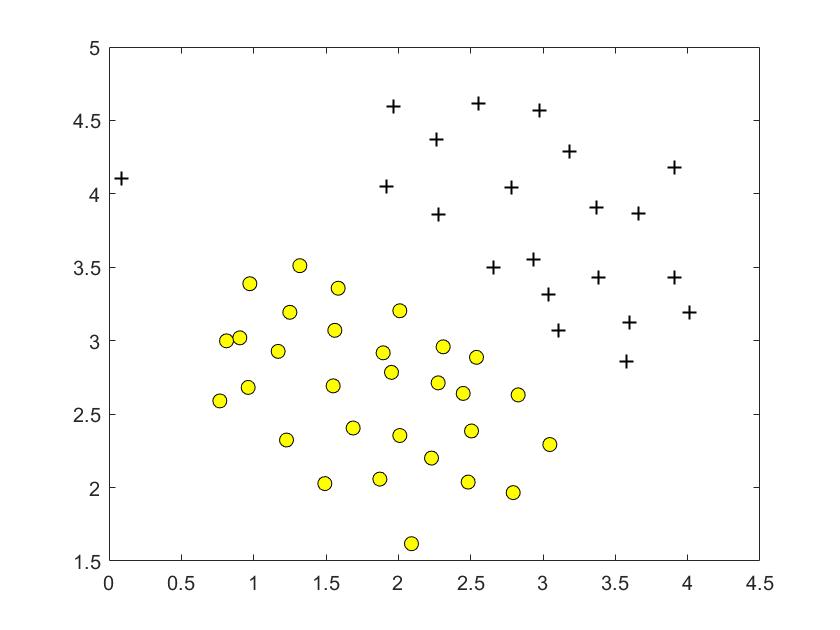
作业任务：本次实验旨在通过支持向量机实现垃圾邮件分类。有ex6和ex6\_spam两个函数，函数中套用了相应的函数，所以需要根据这两个目录函数的指示和实验手册，完成对应的4个部分。

1 Support Vector Machines

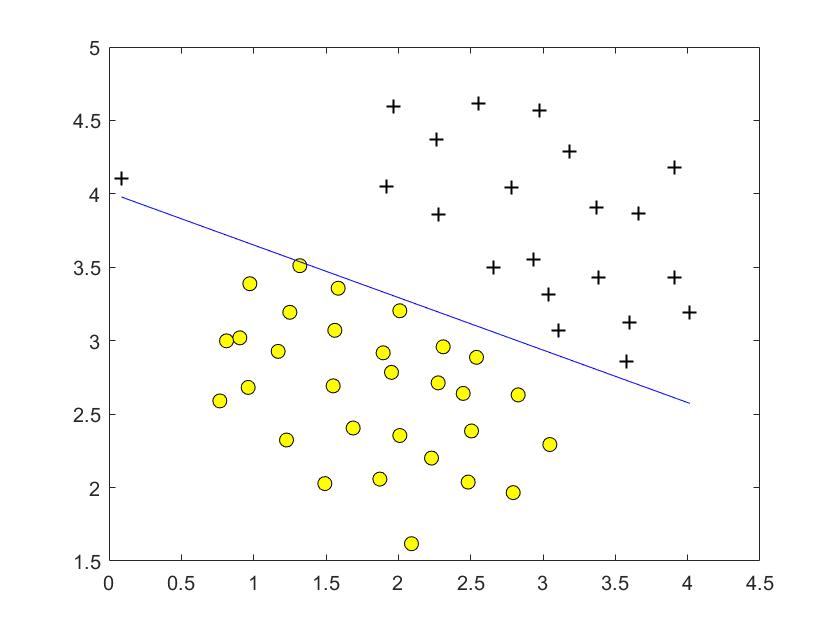
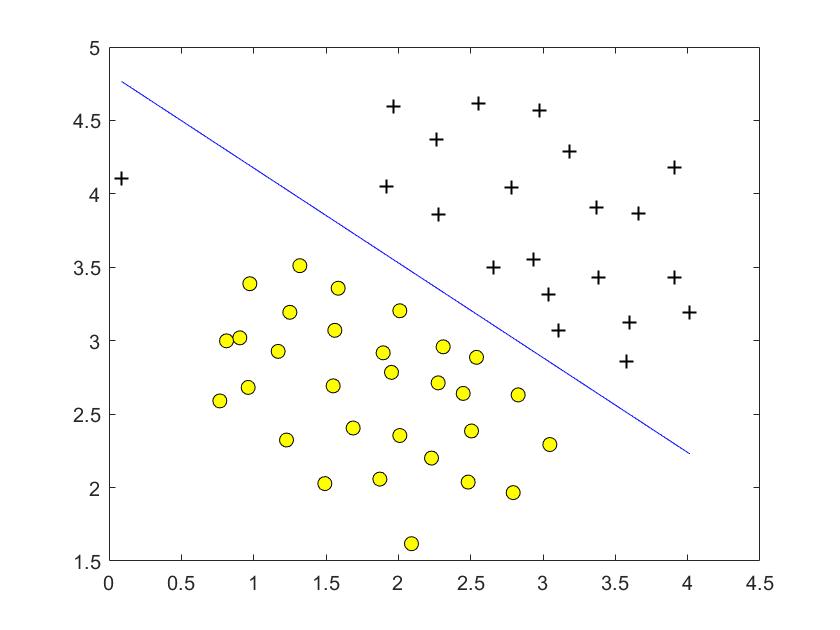
这一部分先实现了2D维度的dataset上的SVMs，这样对SVM有了直观的理解后，在第二部分再应用到垃圾邮件分类上。

1.1 Example Dataset 1

首先是可以分成两部分的dataset，但是我们发现在左上角有一个outlier。在后面可以发现不同的C对outlier的影响。以下是可视化数据：



接下来我们用SVM软件来训练模型，发现C的大小对模型有影响。C越大，模型越倾向于把每一个example都考虑到，于是C的功能类似于1/λ。以下是不同C对模型的影响：



C = 1 C = 100

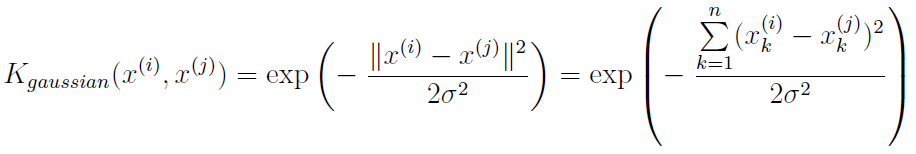
应该注意的是，SVM软件会自动把x加入bias，即x0=1。所以传入的时候不需要手动为x进行加项。

1.2 SVM with Gaussian Kernels

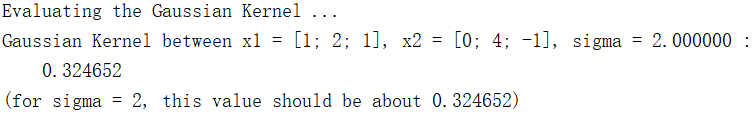
这一部分会用SVM来进行非线性的分类问题，特别用的kernel函数是高斯核函数。

1.2.1 Gaussian Kernel

要实现SVM，首先需要完成高斯核函数，即一种计算example之间距离的相似度函数。高斯核函数公式如下：

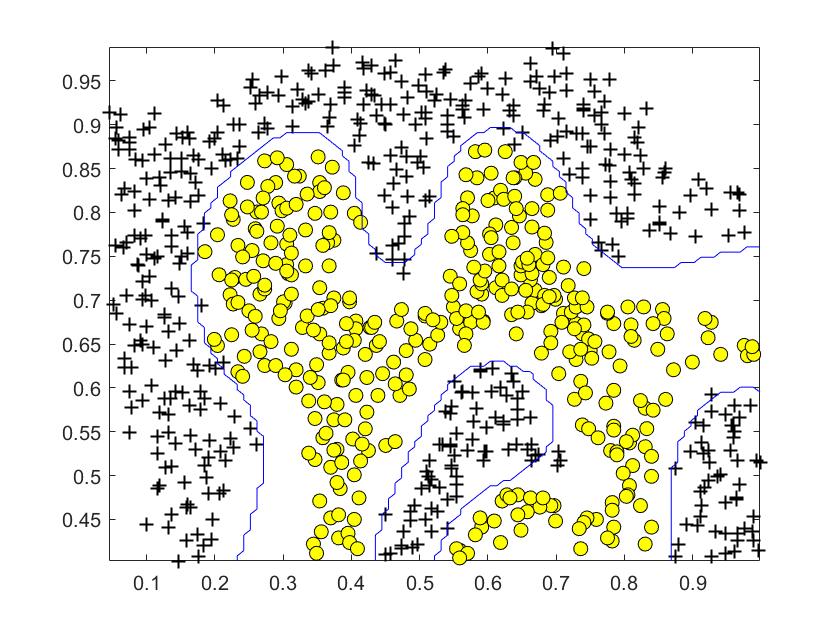
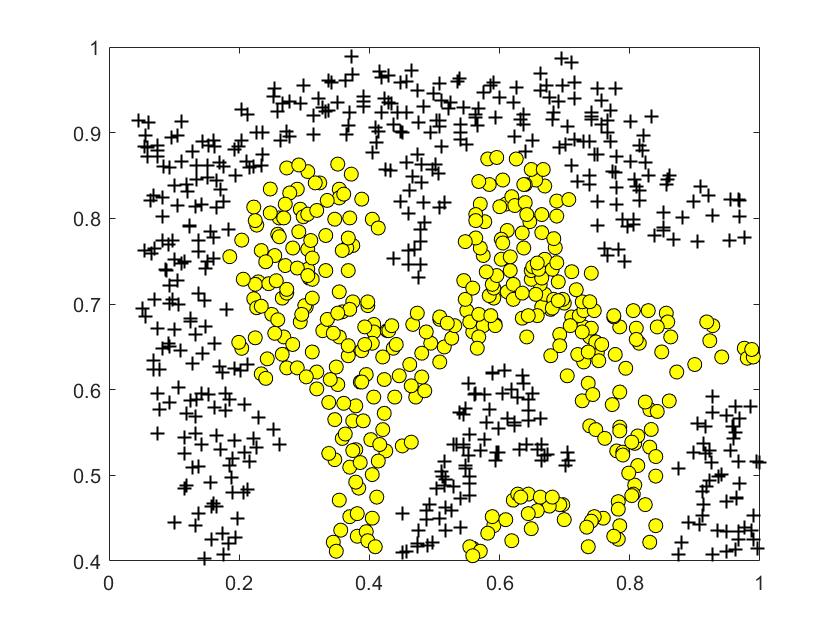


测试结果正确：



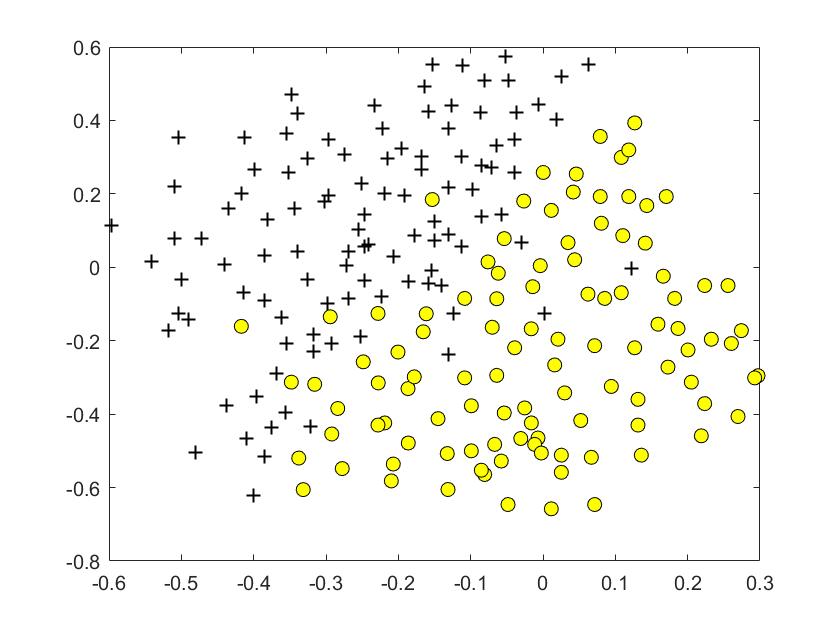
1.2.2 Example Dataset 2

这一部分是一个全新的example。经过可视化，发现无法用直线来分隔两部分，所以需要用到SVM。由于前面已经完善了高斯核函数，所以得到可视化图像和decision boundary如下图：



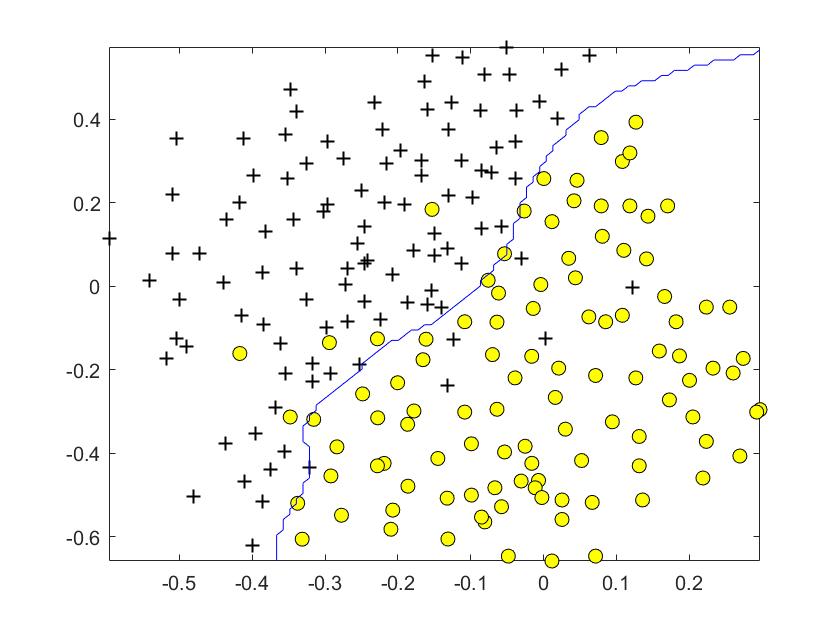
1.2.3 Example Dataset 3

这一部分有另一个dataset，并且基于这个dataset会具体实现基于高斯核函数的SVM：



在ex6的函数中已经用X和y训练好了SVM分类器，我们要做的就是利用Xval和yval来找到最合适的C和σ。这里的思想类似于ex5中寻找最佳λ的思路，只是这里做出简化，可以用mean(double(predictions ~= yval))来代替error。

思路即用C\_vec和sigma\_vec两个数组来表示C和sigma可能取到的所有值。设置一个二层循环考虑两个数组元素的所有组合形式，找出error最小的那个组合，记录下当时的index。最后根据index设置C和sigma即可

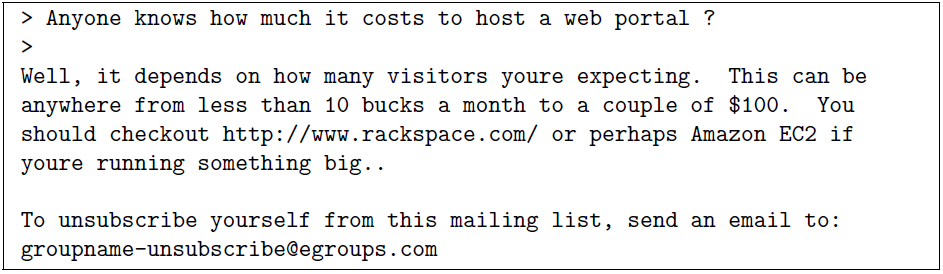


2 Spam Classification

这一部分用SVM实现一个具体的实际问题——垃圾邮件过滤器。我们需要实现给出一封邮件可以区分是否是垃圾邮件，故为一个2类型的分类问题

2.1 Preprocessing Emails

首先我们先对要处理的dataset有一个可视化的认知。以下为我们要处理的邮件：



为了使得算法更加简易可行，我们在processEmail函数中对邮件进行了规范化处理。

1）所有字母转化为小写

2）把HTML去掉

3）所有URL被httpaddr代替

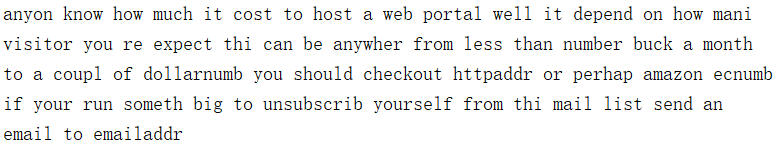
4）所有数字被number代替

5）所有金钱（以$开头）被money所代替

6）对相似单词进行归一化。如discount discounting discounts均被看做discount

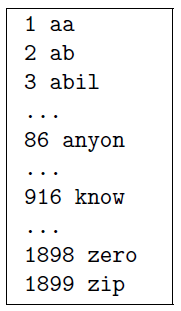
7）去掉所有不是单词的字符，如空格、换行等

经过上述操作没结果如下：



2.1.1 Vocabulary List

我们挑选了在垃圾邮件中出现过至少100次的词汇，共1899个单词组成了词汇表：



有了词汇表后，我们要做的就是将其与我们处理过的邮件进行比照（strcmp），把在邮件中出现过的词汇的index记录下来，得到53个元素。

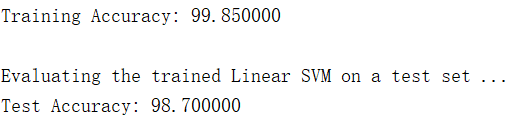
2.2 Extracting Features from Emails

这一部分要做的就是将邮件提取成数组。并且xi={0, 1}。只有当xi在邮件中出现，才可以为1。只需要根据word\_indices中的每个元素的值对x相应位置进行赋值即可。



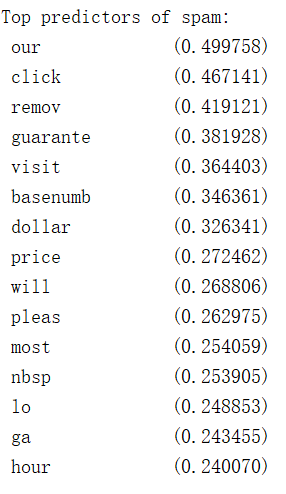
2.3 Training SVM for Spam Classification

这里就用train和test set来进行SVM训练，基于刚刚已经实现的两个函数，得到的accuracy如下：

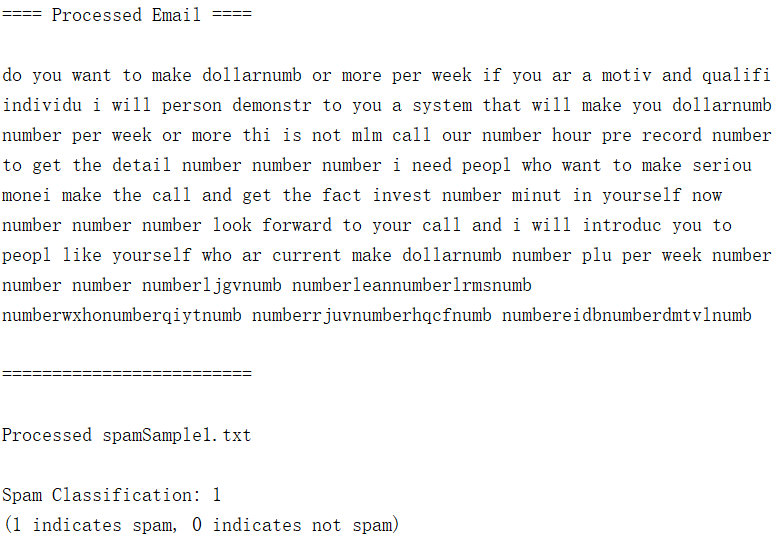


2.4 Top Predictors for Spam

下图体现了哪些词导致SVM判断他是垃圾邮件，即垃圾邮件中最常出现的那些词。



预测结果：



○ 至此实验成功完成

