**第四周-神经网络**

1. 为什么提出神经网络

由于在解决非线性问题时，要预测问题的概率往往需要特别多的特征，最终的多项式也会有很多，并且以几何级增加。所以提出神经网络来解决复杂的非线性分类问题，通过各层不断获取特征，最后得到预测结果。

1. 简单神经网络学习过程



过程如上，第一层为输入层，第二层为隐藏层，第三层为输出层。有如下概念：

代表第j层第i个元素。

代表从j层到j+1层的映射权重控制矩阵，维度为。

代表任意层的第一个元素值都为1，称为偏置节点。

，

预测函数如下：

，是一个值

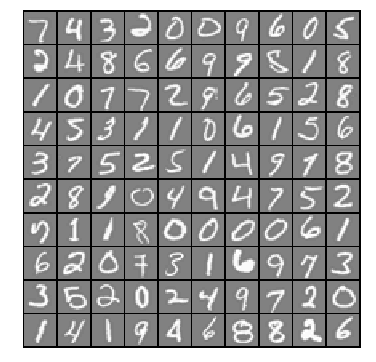
1. 多类分类



每一个都是一个对象的预测结果。

1. 编程作业

一共是5000个样本数据，矩阵维数为5000\*400，是手写数字20\*20灰度图像的5000个样本。显示数据，随机选择100个点进行显示：

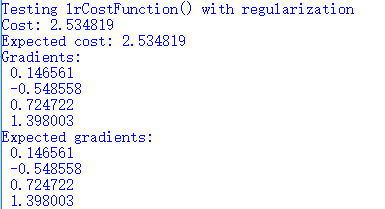


第一问：计算正规化代价函数和普通代价函数，利用



计算正规化代价函数，具体如下：

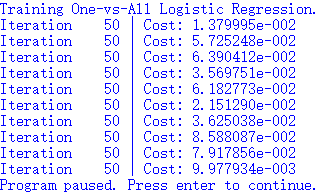
J = ((-y)'\*log(h) - (1-y)'\*log(1-h))./m +(lambda) \* sum(theta(2:end).^2)/(2\*m);



第二问：获取一对多逻辑回归预测模型的参数

计算theta值，具体如下：

[theta]=fmincg(@(t)(lrCostFunction(t,X,(y == c),lambda)),initial\_theta,options);



使用fmincg()函数计算thera值，更加方便，不需要梯度下降方法。迭代50次计算的代价函数结果。Y中的值为1-10的数字（0代表10），Y==c直接将y值设置为0或1。训练得到10个数字的特征值all\_theta矩阵。

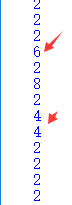
第三问：一对多逻辑回归预测

[k,i] = max(sigmoid(X \* all\_theta'),[],2);

使用sigmoid()逻辑函数作为g(),使用一层隐藏层，生成5000\*10维度的矩阵，矩阵的值代表5000个样本的1-10的概率，对5000个样本数据进行预测，求每一行的最大值（即最可能的数字）的标签作为预测结果。并通过mean(double(pred == y)) \* 100给出预测的准确率。实验结果如下：

Y值（正确值）为10（500）、1（500）、2（500）、3（500）、4（500）、5（500）、6（500）、7（500）、8（500）、9（500）。

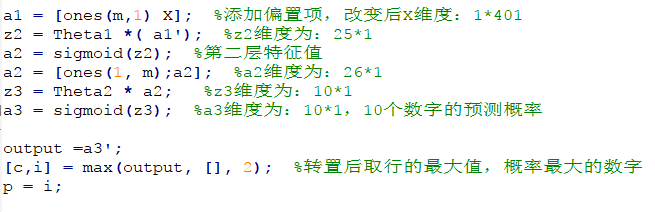
Pred值（预测值）:预测率为95.02。出现如下预测错误：



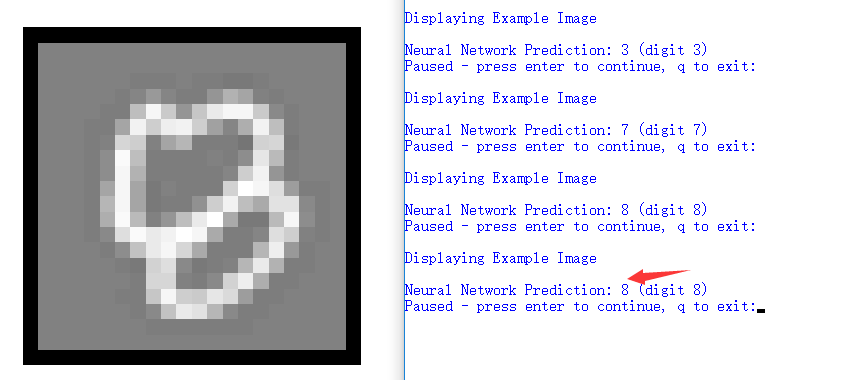
第四问：数字识别：

当特征比较多时，使用神经网络算法来完成，通过加载ex3weights.mat里的值，可知theta1的维度为25\*401，theta2的维度为10\*26，辨别数字为0-9，使用25层隐藏层。

代码解析：



实验结果：



学习总结：

本周学习主要是神经网络的基础知识，学习了简单神经网络的学习过程、以及神经网络能解决那些问题。通过学习存有一下疑问：（1）thera参数矩阵如何获取（2）隐藏层几层能合适预测结果，或多或少。对于多个输出结果进行预测时，求其最大可能性，如识别数字。本周的编程作业，作业1主要是代价函数->训练得到theta参数矩阵->预测并求最大可能，作业2主要是给定theta参数矩阵->预测单个数字并显示结果。通过作业提示、查阅资料和请教学长，最后能够理解代码并正确提交代码。