|  |
| --- |
| **Github账号：https://github.com/ZmA0/Coursera\_machine\_learing\_exercise** |
| **实验题目:逻辑回归中多类分类的编程实现练习** |
| **实验摘要：**   1. 实验目的：   利用matlab或者octave，掌握对机器学习中的逻辑回归中多类分类的情况进行编写。进行代价函数及偏导数的归一化运算和优化算法等的使用。   1. 实验要求：   利用matlab或者octave的数学编程语言进行书写，按照代码文件中的要求编写并提交文件，使得程序运行通过并且得到足够的分数即为通过。   1. 代码文件要求：   需要编写的文件为  IrCostFunction.m,  oneVsAll.m,  predictOneVsAll.m,  predict.m |
| **题目描述**   1. IrCostFunction.m:   这道题的instructions部分说明我们需要计算两个值，一个是逻辑回归中的代价函数J，还有一个是该代价函数的偏导数也就是。不过同时也要注意这两个值都是要经过归一化处理的。Hint部分告诉我们一个是要利用向量化处理数据，另一个就是还告诉我们在进行归一化处理时，可以先利用一个中间变量将theta（1）置0，这样一来进避免了多惩罚项的错误了  2.oneVsAll.m  这道题需要我们完成多类分类情况中输出情况为多个类别时theta不同值的返回函数。题目中，all\_theta代表不同情况下theta最优值的不同解的矩阵，每行为每种情况下的theta值；然后hint部分给我们示例了fmincg算法的使用。我们需要利用这个优化算法算出每个情况下theta的最优值，并将其储存在all\_theta矩阵中。  3.prefictOneVsAll.m  这道题需要我们利用第二个的结果来进行多类分类结果的处理。通过逻辑回归中的S型函数将all\_theta的值与X进行运算，也就是得到一系列假设函数H的值。这些H的值是储存在一个矩阵中，然后对矩阵的行为单位进行操作得到最大值即最可能值，然后将最可能值的坐标储存在一个列向量中则满足题意。Hint部分告诉我们可以利用max函数进行操作。  4.predict.m  这道题需要我们进行神经网络的初步运算。给出了两个权重矩阵Theta1和Theta2，也就是说这个网络的结构只有一个输入层X一个隐藏层a和一个输出层。与上一题相似的地方也是多累分类，即神经网络输出为一系列列向量，同样需要max函数来进行输出值的处理，返回最大值所在的坐标列向量即可 |
| **实验过程**   1. IrCostFunction.m:   根据上面的题目描述可以知道，这道题的具体过程与上一次实验中的最后一个文件基本相同，同样是编写归一化的代价函数及偏导数，我们同样按照公式编写即可。不过这里给了我们提示可以利用中间变量将theta（1）置0，这样一来就不需要我们上次那样将平方项加上之后再减去，因为这样的运算过于笨拙，而且的确容易出错。因此，代码如下：（代价函数保持上次方法，偏导数利用这次的方法）  J1=(-1)/m\*sum(y.\*log(sigmoid(X\*theta))+ (1-y).\*log(1-sigmoid(X\*theta)));  J2=lambda/(2\*m) \* (sum(theta.^2)-theta(1)^2);  J=J1+J2;  grad1= 1/m \* X'\*(sigmoid(X\* theta) - y);  temp=theta;  temp(1)=0;  grad2= lambda/m \* temp;  grad=grad1+grad2;  2.oneVsAll.m  这道题题目已经给了我们优化算法的使用示例，我们所做的也就是按照示例敲一遍即可，因为这个fmincg算法是已经封装好的，我们直接调用即可。这道题还有一个要求就是这是在多类分类前提下完成的，而且每种情况的theta值需要储存在一个新矩阵的一行中，原theta是一个列向量。对于这样我们可以编写一个for-loop，然后在第i次循环结束的地方，将这次循环产生的theta值以转置的方式赋给新矩阵的第i行（hint部分说明可以利用theta（:）将其值展开，不过好像这里不需要这么做）。综上分析，代码如下：  initial\_theta=zeros(n+1,1);  options = optimset('GradObj', 'on', 'MaxIter', 50);  for i=1:num\_labels  [theta]=...  fmincg(@(t)(lrCostFunction(t,X,(y==i),lambda)),...  initial\_theta,options);  all\_theta(i,:)=theta';  end  的变小而变大；当实际的  y=0  且  h  θ  也为  0  时代价为  0  ，当  y=0  但  h  θ  不为  0  时误差随着  h  θ  的变大而变大。    将构建的  Cost(h  θ  (x),y)  简化如下： |
| 3.predictOneVsAll.m,  这道题其实道理并不复杂。多类分类中处理方法也就是将每种情况先当做二元分类进行运算，然后将每种分类都运算一遍将各自得到的H值最大的那个当做分类后的结果。不过我们这里需要用到max函数，这个函数用法有很多，这里需要用到三个参数的情况，也就是将矩阵的行为单位将每行的最大元素挑选出来，同时也要返回两个值，一个是最大元素的列向量，另一个是最大元素对应位置的列向量。表示为[maxValueVec,indexVec]=max(Matrix,[],dim);  这样做的原因是我们利用S型函数得到结果矩阵，是包括所有类别的结果的矩阵，维数是m \* num\_labels 也就是说每行的结果都是一个样本对应每种类别的分类情况概率大小，最大的即为该样本输出最可能值。综上所述，代码如下：  res=sigmoid(X\*all\_theta');  [r,i]=max(res,[],2);  p=i;   1. predict.m   这道题是对神经网络做一个初步模拟，通过上面的分析描述可以知道我们主要需要计算隐藏层的激活单元a矩阵和最后输出矩阵的结果的处理，由于这里也是多类分类的情况，也就是说输出结果我们同样需要进行上一题的计算方法，也就是说利用max函数对结果矩阵的每一行进行挑选，才能满足题意。这里需要注意的是对激活单元a需要加上偏倚单元a0=1，否则会出现计算错误，同时还应该注意矩阵的维度，比如最后的结果矩阵维度是num\_labels \* m维的，需要进行转置后才能进行max运算，这样p才是符合要求的列向量（当然也可以最后对p进行转置，总之要注意max参数不同以及返回向量的形式）。综上代码如下：  X\_new=[ones(m,1) X];  a=sigmoid(Theta1 \* X\_new');  a=[ones(1,size(X\_new,1));a];  res=sigmoid(Theta2 \* a);  b=res';  [c,i] = max(b,[],2);  p=i; |
| **实验总结**  第三次编程个人觉得比前两次的编程规模以及实用性都要强了不少，而且在思维上要求更高，其中多类分类的一些操作以及和max的结合都需要我在matlab命令行里面一个个参数的看和观察，再结合题意才能大致明白题目要我们干什么。这次编程中出现了许多困难，比如在进行第三个编程实现时，对max函数的理解刚开始不能很好的与题意结合，查过文档，并且差了许多东西但是仍然没有理解对行最大的操作与多类分类结果的联系。后来有看了一遍课程笔记，然后在命令行里把每个参数都以小规模模拟了一遍，才明白这道题里用max的意图。这也确实认识到了在编程后面需要有很好的算法思想的支持，本来比较复杂的题意，有了很好的向量化思想的支持，就可以简化到只用一个函数解决问题。 |
| **参考文献**  **<https://wenku.baidu.com/view/f328b62b69dc5022abea0068.html>**  （同样参考了这个笔记）  <http://www.ilovematlab.cn/forum.php?mod=viewthread&tid=206757>  （max函数的用法）  <https://www.zhihu.com/question/34360115>  （知乎一个相关问题） |