**人工智能原理实验**

**实验四 基于神经网络的模式识别实验**

**组长：**周俊峰

**组员：**周俊峰

1. **实验目的**

理解神经网络的结构和原理，掌握反向传播学习算法对神经元的训练过程，了解方向传播公式。通过构建网络模式识别实例，熟悉前馈网络和反馈网络的原理和结构。

1. **实验内容**

1.设计神经网络模型，实现数字图像的分类。见书本217页例8.1。

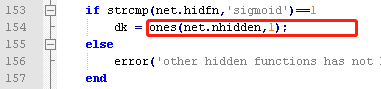
2.已给定的程序可以直接运行main / main4student命令并观察结果（但程序并不正确）。

3.给定的程序有三个文件，main4student.m / loadimages.m / plotNum.m ，其中main4student.m覆盖了整个模型的所有计算代码。

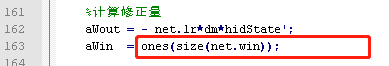
1. **实验要求**

**（注：为避免表格重复赘述，我将内容直接填入要求表格中）**

1. 理解书本215页的推导，在main4student.m文件中的netTrain函数内，**修改dk以及aWin的计算，使得程序能够正确运行**。dk和aWin的数学定义分别在书本215页的（8.18b）和（8.16b）

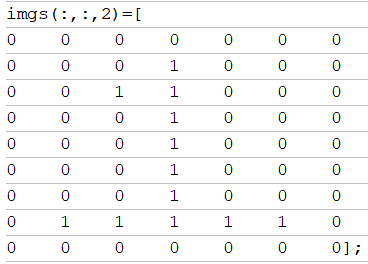
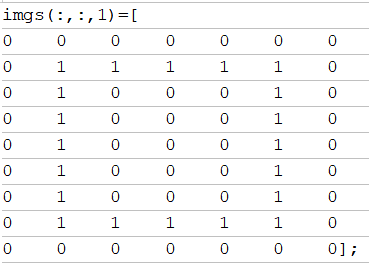


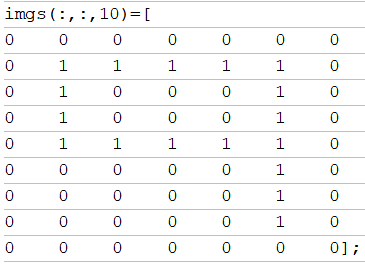
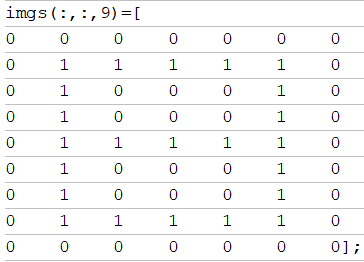
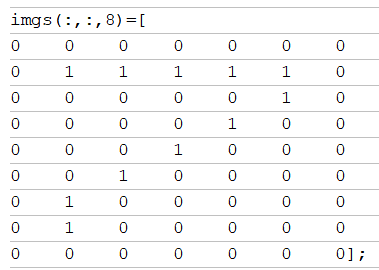
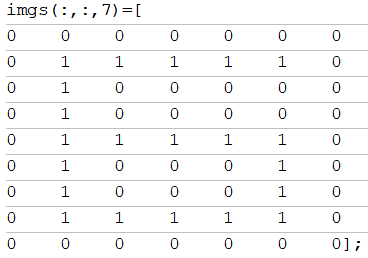
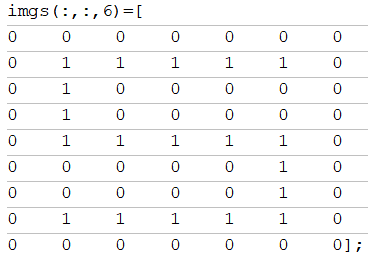
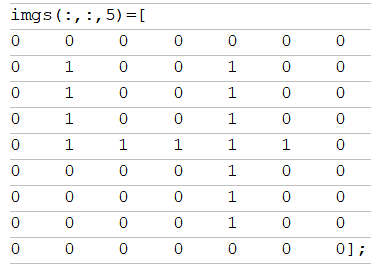
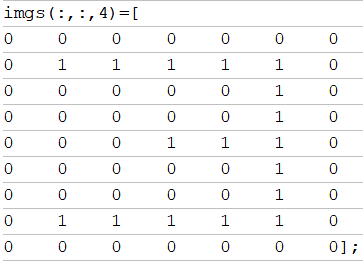
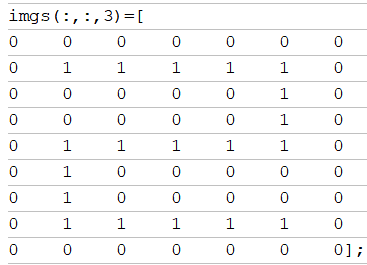
**dk =hidState.\*(1-hidState).\*(net.wout'\*dm);**



**aWin = - net.lr\*dk\*input';**

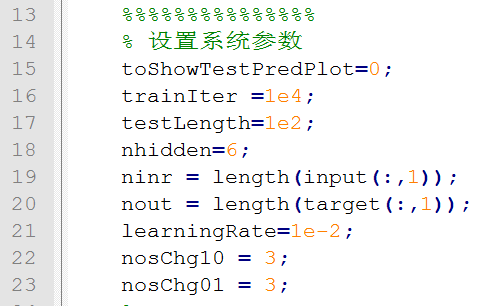
1. loadimages.m文件目前仅实现了0/1两个数字的保存，请依照书本图8.5编写imgs三维数组的其他部分，使其保存0-9十个数字图片；可以使用命令plotNum( )来显示数字的具体图片,如plotNum(imgs(:,:,2))。





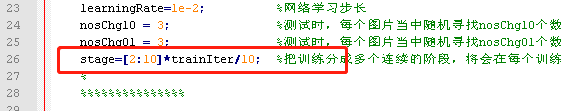
3. 改写main.m/ main4student.m的内容，使其从代码上实现0-9等十个数字的分类；

4. 保持代码中的系统参数（大概代码15-23行）不变的情况下，把nosDigits从2改变成10，观察系统被训练1e4次之后，下面的量的变化，并填写表格一。



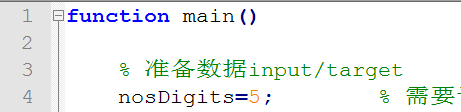
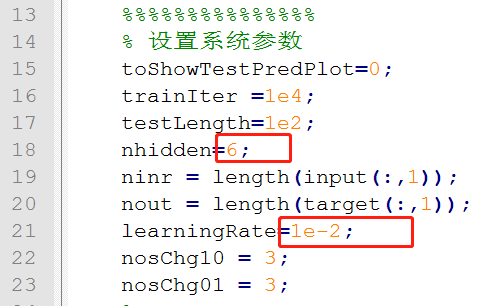
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **表格一** | nosDigits | | | | | | | | |
| 变量 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| errTrn1000 | **0.11374** | **0.099967** | **0.085694** | **0.075969** | **0.078549** | **0.074955** | **0.070261** | **0.079392** | **0.067088** |
| errTestMean | **0.14009** | **0.11719** | **0.099979** | **0.083438** | **0.087752** | **0.081472** | **0.075564** | **0.083694** | **0.071736** |

1. 设置nosDigits=5，并保持代码中的系统参数（大概代码15-26行）其他参数不变的情况下，更改参数stage=[2:10]\*trainIter/10; 使训练分成2~10等九个阶段。运行程序，记录下每个阶段的error变化。



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **表格二** | Stage | | | | | | | | |
| 变量 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| errTrn1000 | **0.26878** | **0.21924** | **0.17513** | **0.14127** | **0.11826** | **0.10226** | **0.091058** | **0.082408** | **0.075944** |
| errTestMean | **0.2518** | **0.20959** | **0.17368** | **0.14479** | **0.12634** | **0.11191** | **0.099845** | **0.093173** | **0.085068** |

6. 设置nosDigits=5，并保持代码中的系统参数（大概代码15-26行）其他参数不变的情况下，还原stage=[10:10]\*trainIter/10; 并更改nhidden（网络隐单元层神经元个数）以及learningRate（网络学习步长）的数值，观察**errTrn1000**变量的变化，并填写表格三。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **表格三** | | nhidden（网络隐单元层神经元个数） | | | | | | | | | | | | | |
| 2 | 3 | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | 10 |
| Learning  Rate  网络学  习步长 | 1 | **0.010341** | **0.0078064** | | **0.0072557** | | **0.0064898** | | **0.005579** | | **0.005243** | | **0.0048471** | **0.0046558** | **0.0043248** |
| 1e-1 | **0.036396** | **0.026052** | | **0.023339** | | **0.020697** | | **0.019521** | | **0.017609** | | **0.016088** | **0.015688** | **0.014783** |
| 1e-2 | **0.14414** | **0.13685** | | **0.098453** | | **0.10252** | | **0.076763** | | **0.069007** | | **0.069543** | **0.060475** | **0.061843** |
| 1e-3 | **0.37439** | **0.34343** | | **0.32433** | | **0.31325** | | **0.29582** | | **0.2938** | | **0.28894** | **0.28284** | **0.2788** |
| 1e-4 | **0.4899** | **0.48812** | | **0.49079** | | **0.48926** | | **0.4784** | | **0.47431** | | **0.45002** | **0.4521** | **0.45602** |

7. 代码中的激活函数采用了书本211页（8.6）公式的sigmoid函数形式，并a=1。修改a的数值以及相应代码（包括test和train两部分）并观察结果的变化。另外，尝试把激活函数修改成为线性输出(linear)，即 f(x) = x 。把以上修改实现后运行观察到的数据填充到表格四中。激活函数改为线性输出(linear)时，需要把createNet函数中修改net.hidfn和net.outfn两个量，并修改netTrain函数代码。

其中**nosDigits=5;nhidden=6;stage=[10:10]\*trainIter/10;learningRate=1e-2**;

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表格四** | a数值(sigmoid函数) | | | | | | linear |
| 变量 | 0.1 | 0.2 | 0.5 | 1 | 2 | 3 |
| errTrn1000 | **0.31693** | **0.29569** | **0.15688** | **0.08092** | **0.02662** | **0.012521** | **2.9855e-17** |
| errTestMean | **0.31926** | **0.29772** | **0.18584** | **0.11269** | **0.096279** | **0.069612** | **0.14871** |

1. **实验总结**

1.随着训练次数的不断增多，准确率越来越高，最终趋于某个定值。

2.激活函数将极大的影响训练成果