

实 验 报 告

实验名称_用于字符识别的三种神经网络的性能对比_

课程名称_____神经网络与深度学习_____

学生姓名:韦璐

学 号:PB16000702

实验日期:2020-12-25

中国科学技术大学

实验报告要求：

实验三：用于字符识别的三种神经网络的性能对比

- 1) 感知器
- 2) BP 网络
- 3) Hopfield 网络

可以看看 MATLAB 环境下一些老版本中的演示程序

一、 描述实验概要和实验目标：

二、 实验过程

1、实验设计

①三个神经网络的输入、输出矢量及问题的阐述；

②分别给出三个神经网络的结构

2、训练样本：单独无噪声时/无及有噪声都有时（选择加噪声的范围，对程序已经给定的噪声，进行扩大，最大噪声需满足在 0.2-0.5 之间，可多次加入不同范围的噪声进行多次实验）

3、测试样本：选择需要识别的字母（或整个字母表），测试当被识别的字母在含有指定噪声范围噪声情况下，输入 100 次测试的正确识别率，画出验证实验结果图（重点在三个网络的性能对比图上。把三个网络的识别性能画在一个图中进行对比）。

4、对网络进行调试

根据实验目标，选择不同优化算法、不同学习率等，继续实验，研究其对实验迭代次数和识别正确率的影响。

三、实验分析和实验结论

- 1、根据实验结果，对三种网络的抗噪性能进行对比分析；
- 2、根据实验结果，探究所选变量对实验结果的影响并进行分析。
- 3、写出对本次实验的心得体会。

四、提交电子版实验报告和代码，[压缩打包提交 lwryjj@mail.ustc.edu.cn](mailto:lwryjj@mail.ustc.edu.cn)

一. 描述实验概要和实验目标：

测试感知器，BP 网络和 Hopfield 网络的模式识别能力以及对噪声的容忍程度。

二. 实验过程

1. 实验设计

a) 三个神经网络的输入，输出矢量及问题的阐述：

- 感知器网络、BP 网络：

输入——三十五像素的输入矢量，是把 7*5 的二十六个字母拉成长条，测试的时候用的是加入随机噪声的字母

输出——二十六阶对角矩阵

问题——训练网络能够实现对二十六个字母的模式识别

- Hopfield 网络：

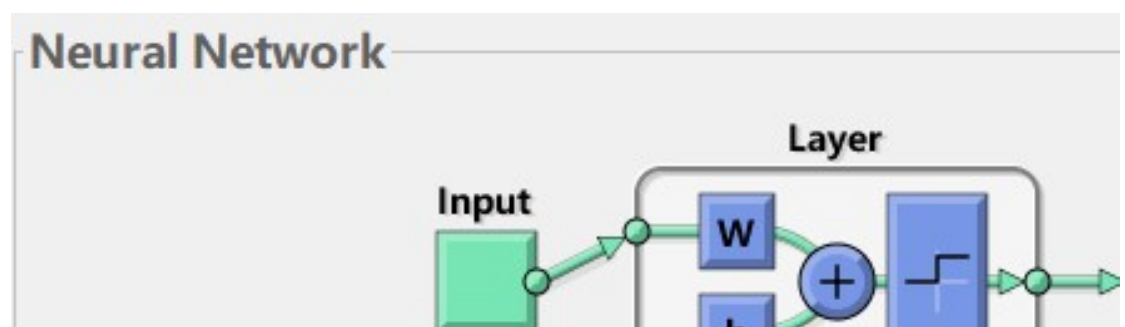
输入：三十五维向量

输出：三十五维向量

问题：二十六个字母的模式识别

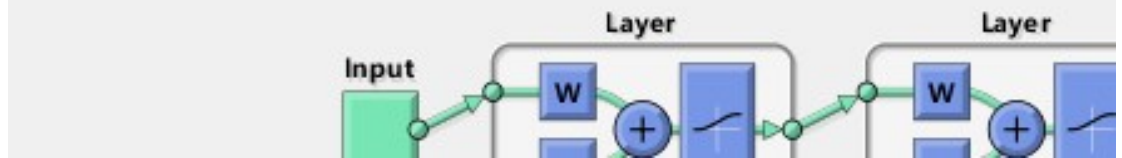
b) 分别给出三个神经网络的结构

- 感知器网络：



- BP 网络：

Neural Network

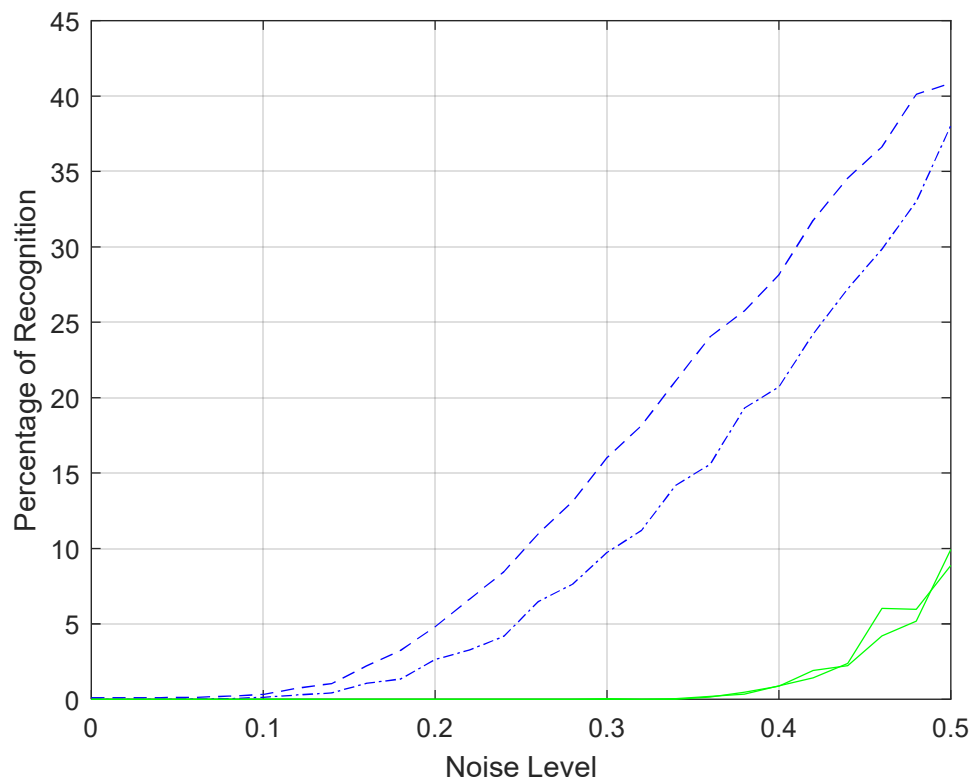


- Hopfield 网络:

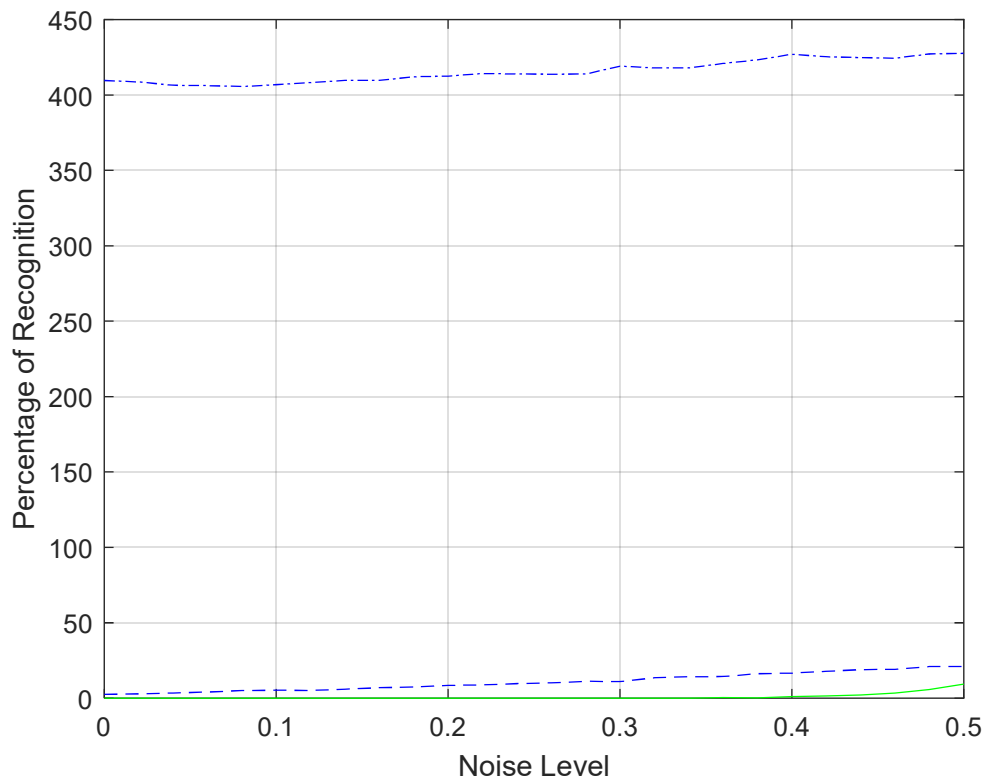
三十五个神经元的网络

2. 训练样本: 这里所指的含噪训练在前面的实验中都已经实现 0-0.5 的含噪训练, 而 hopfield 网络因为吸引子给定以后结构就给定了, 无需进行训练。
3. 测试样本: 选择需要识别的字母 (或整个字母表), 测试当被识别的字母在含有指定噪声范围噪声情况下, 输入 100 次测试的正确识别率, 画出验证实验结果图 (重点在三个网络的性能对比图上, 把三个网络的识别性能画在一个图中进行对比)。

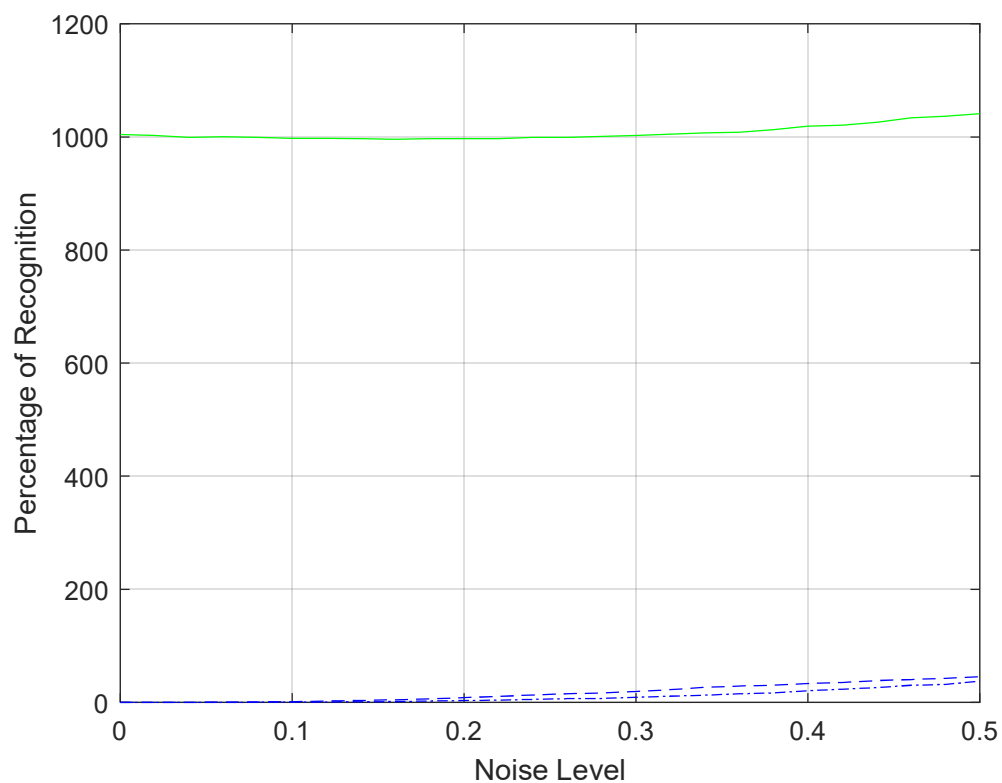
这里我们对感知器网络和 BP 网络都采用的含有 0-0.5 随机噪声的训练数据进行训练, 然后对三个网络都用某个字母进行 100 含有 0-0.5 随机噪声的测试。



关于这张图, 我一开始想在本次实验的 `duibi.m` 文件中一次实现这三幅图的绘制, 但是 hopfield 网络的组织过程中就包括了对于 `alphabet` 这个数组的修改, 如果直接什么也不管的话就会如下图:



如果在hopfield网络后面加上一句`alphabet(find(alphabet==-1))=0;`
结果就会如下图:



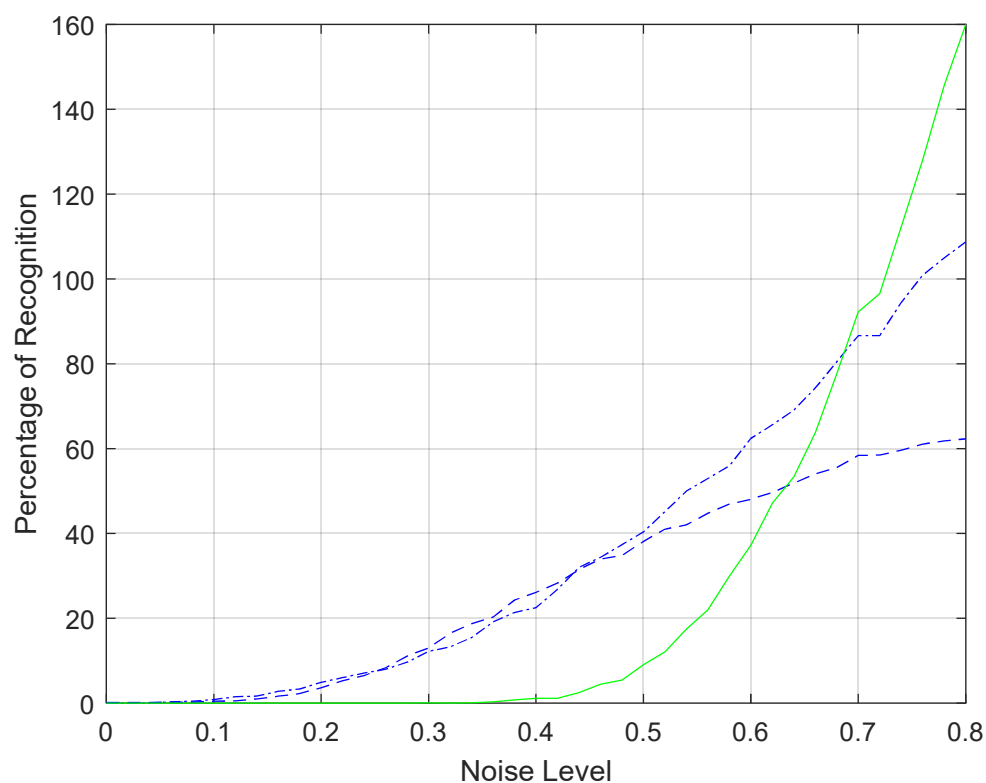
也

尝试了定义一个(35*26)的零矩阵,但是没有用,总之就是在这里滞留了两三个小时,等寒假我再继续研究这个问题,最后我能画出第一张图是我在绝望之下删除了 `duibi.m` 文件中最后 `plot` 中关于 `network2` 的语句,然后直接运行,打算先贴了 `network1` 和 `network3` 的图再运行 `hopfield`

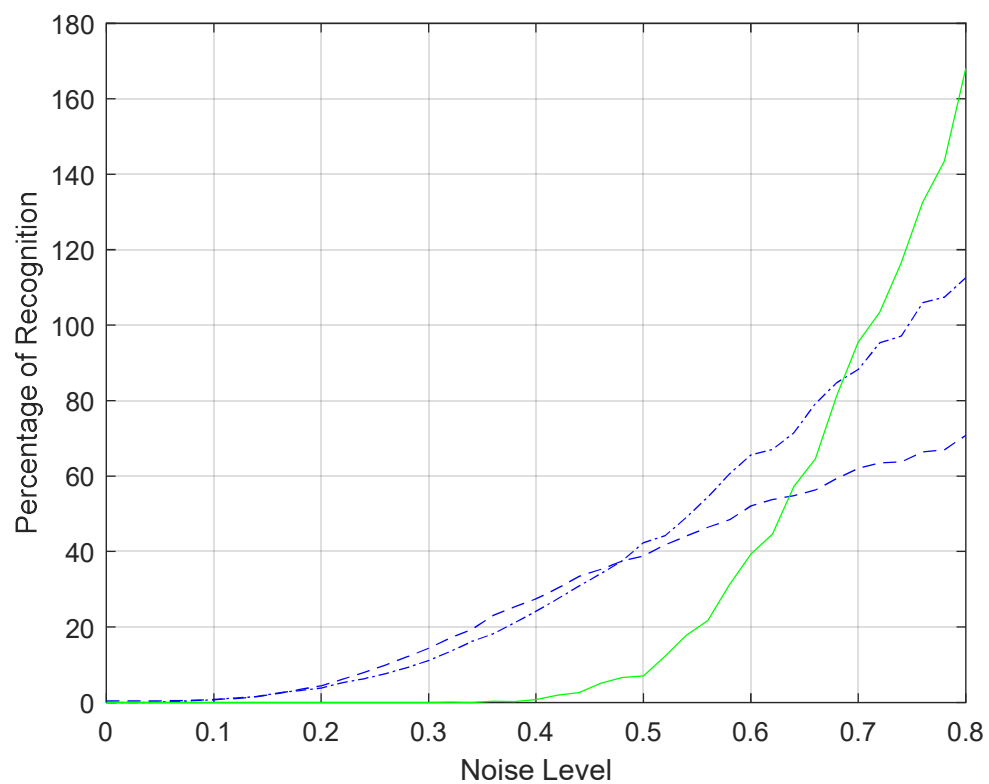
网络，结果我运行 **hopfield** 的时候忘记关 **figure1** 了，却发现它们自动画在同一张图里面，于是我终于可以把第一张图交上来了。

4. 对网络进行调试：根据实验目标，选择不同优化算法、不同学习率等等，继续实验，研究其对实验迭代次数和识别正确率的影响。

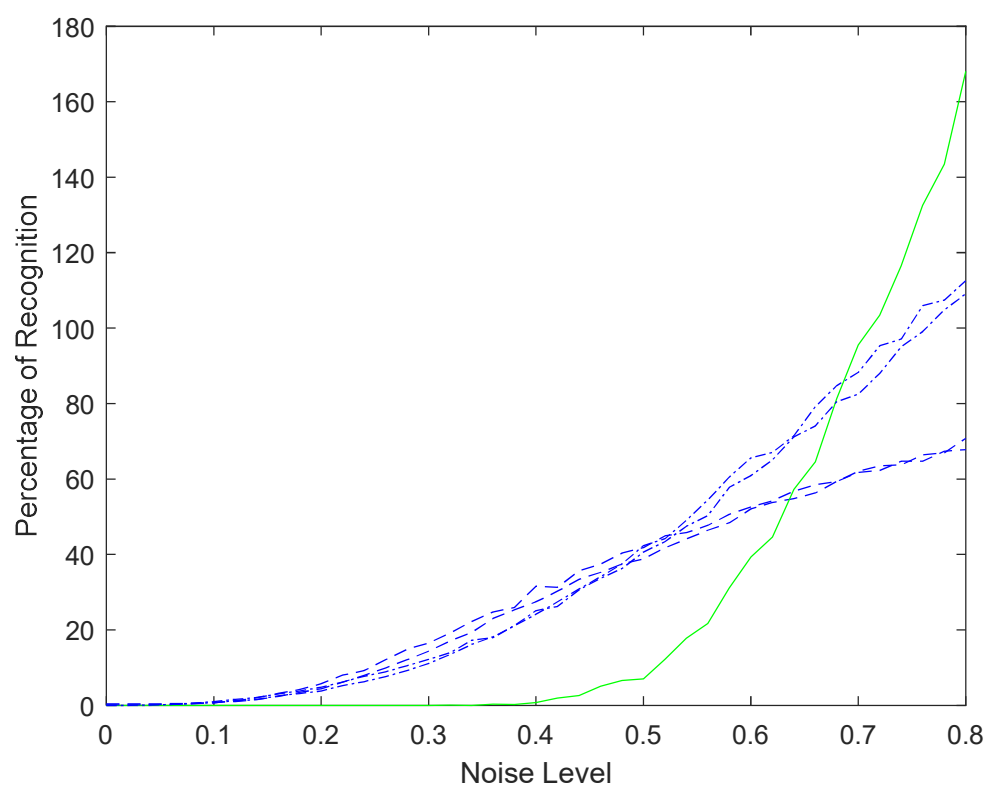
我们上面的实验中 BP 网络是采用的附加动量法进行的训练，单层感知器网络的训练方式只有一种，而 **hopfield** 网络不需要训练，所以我们只改变 BP 网络，现在我们把学习方法改为变速率学习法进行训练，这里变速率法的数值是系统自动的数值。



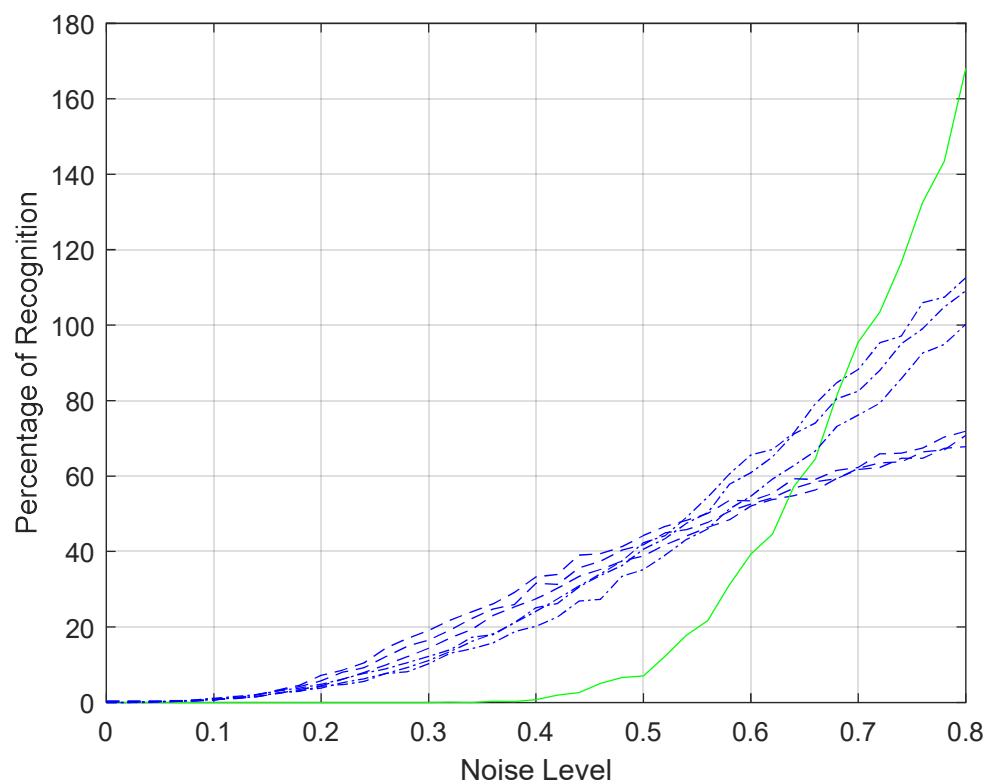
下面将变速率学习法增长的学习率改为 1.01 进行训练：



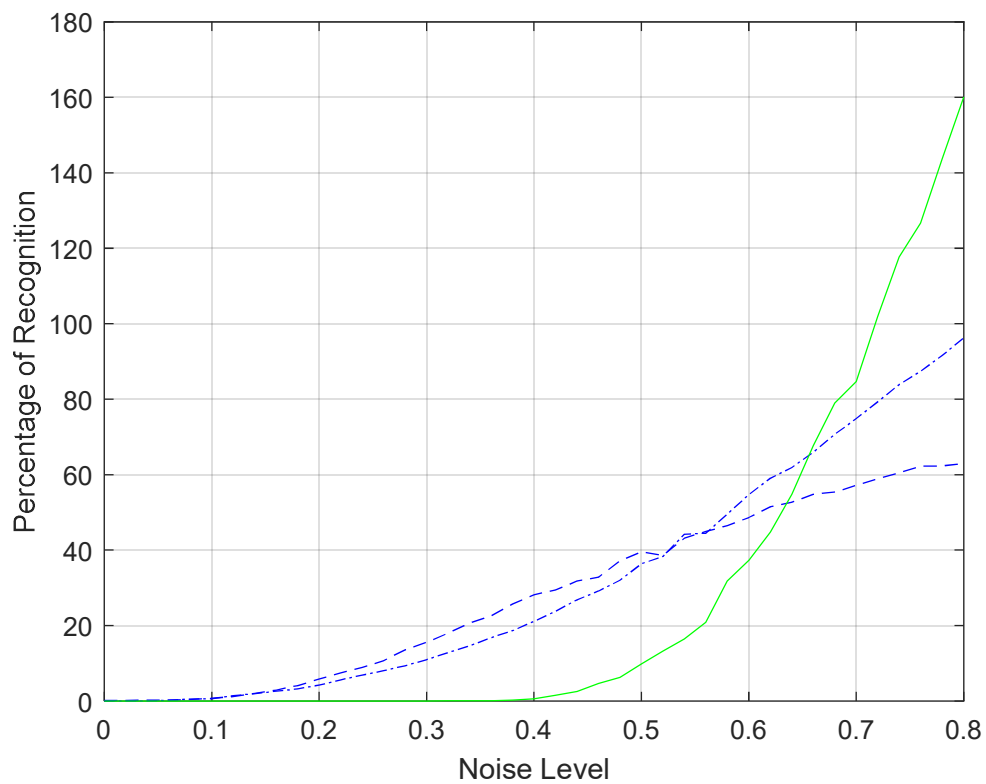
再改为 1.05 运行一次:



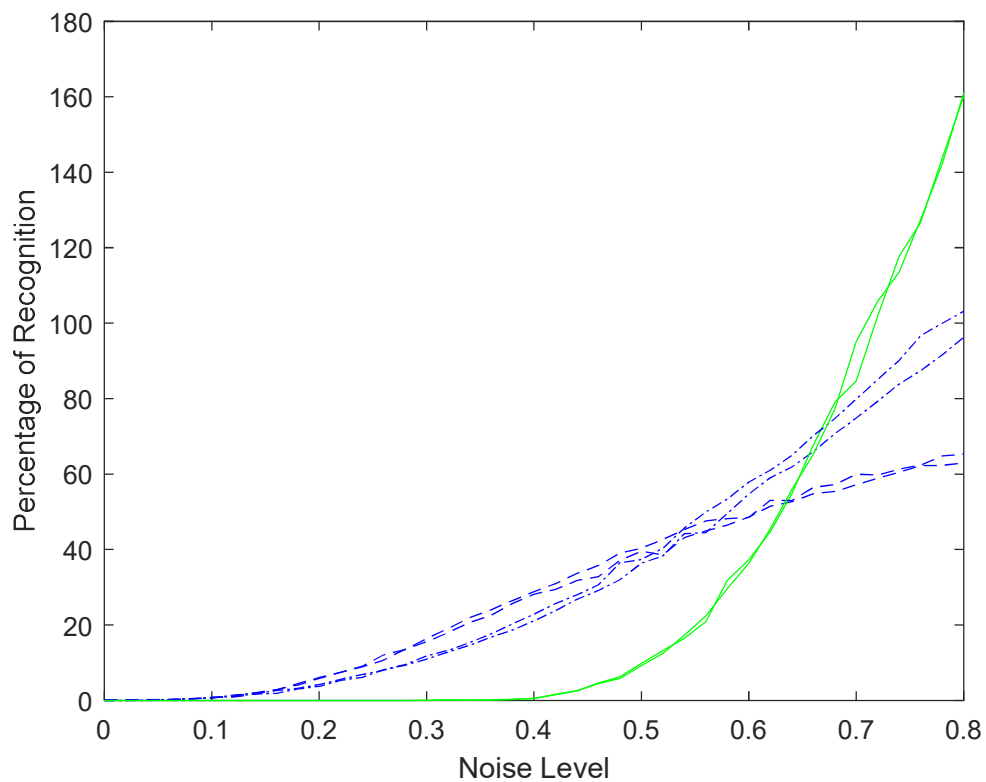
再改为 1.09 运行一次:



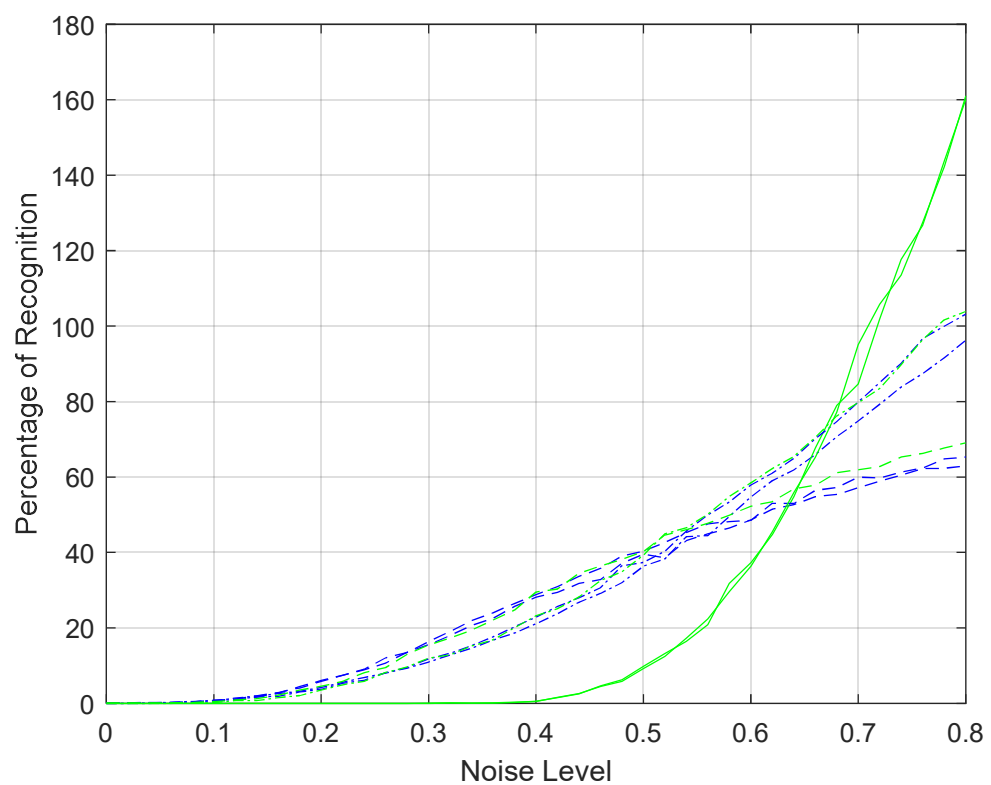
可以看出值越大整体误差越小。
下面修改 lr_dec 为0.6进行观察：



还原为0.7：



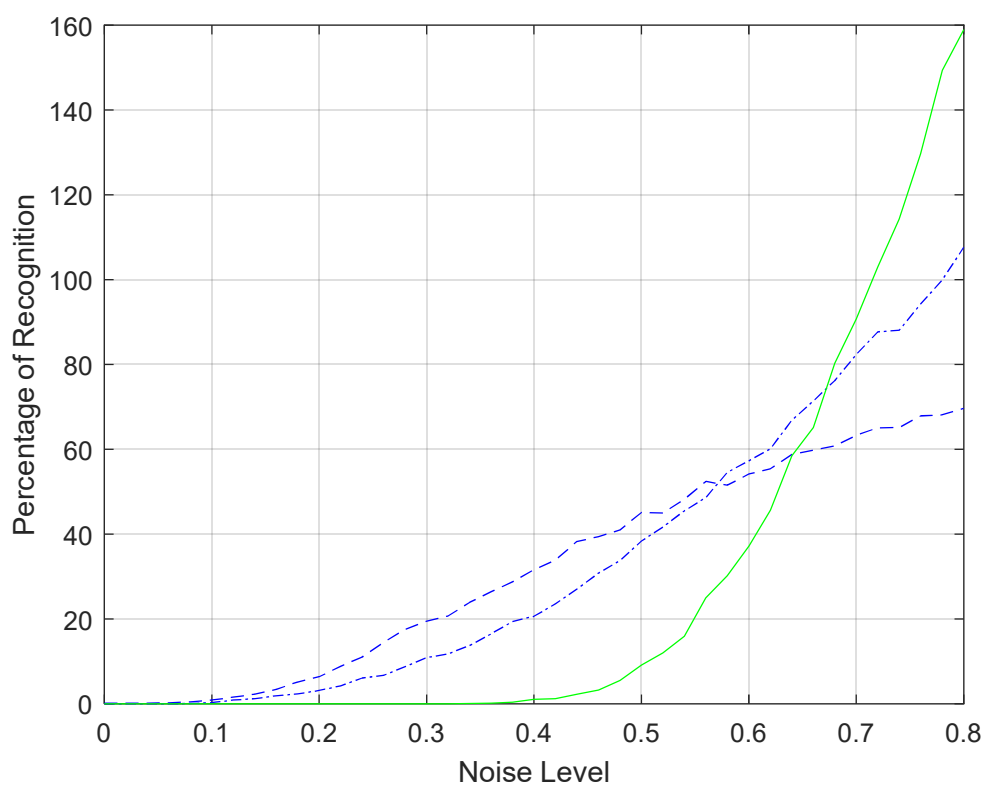
进一步 0.8:



可以看出 lr_dec 越大 BP 网络整体误差越大。

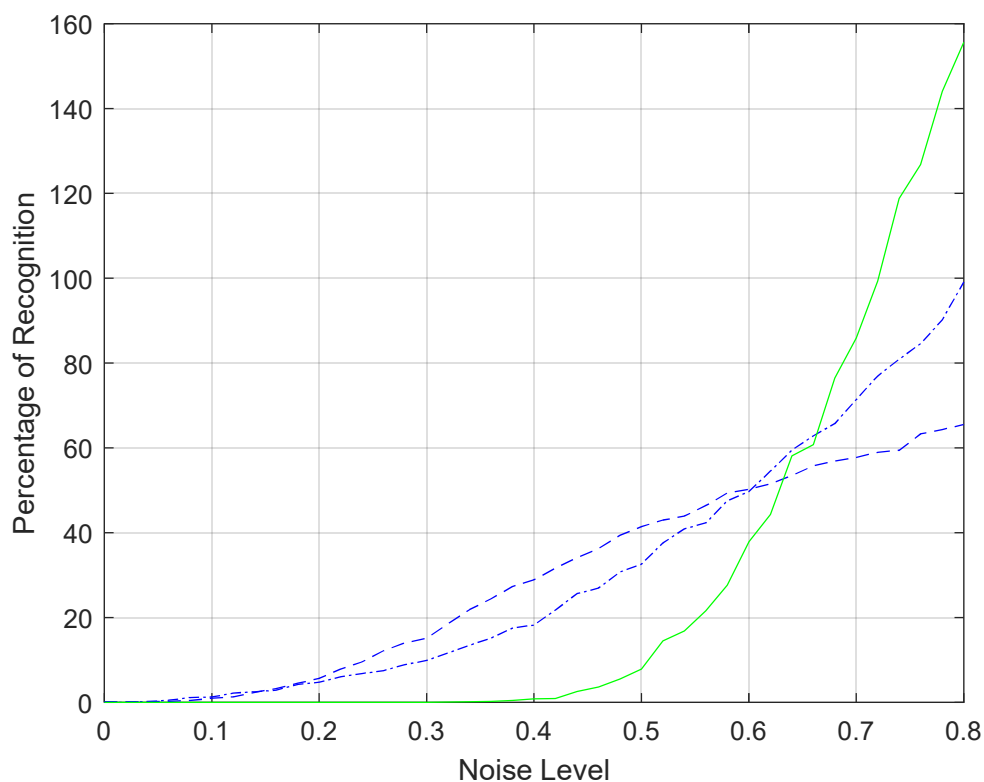
可以看出变速率学习法训练的 BP 网络性能也是在一定误差以后才优越于附加动量法训练的网络，但是两者之间的差值非常小，产生反转也比较早，而且在之后误差变大的时候曲率越来越小。

下面改变附加动量法的固有参数进行训练,将附加动量因子由 0.9 改为 0.8:



可以看出和 0.9 相比没有什么变化。

下面 改附加动量为 0.95:

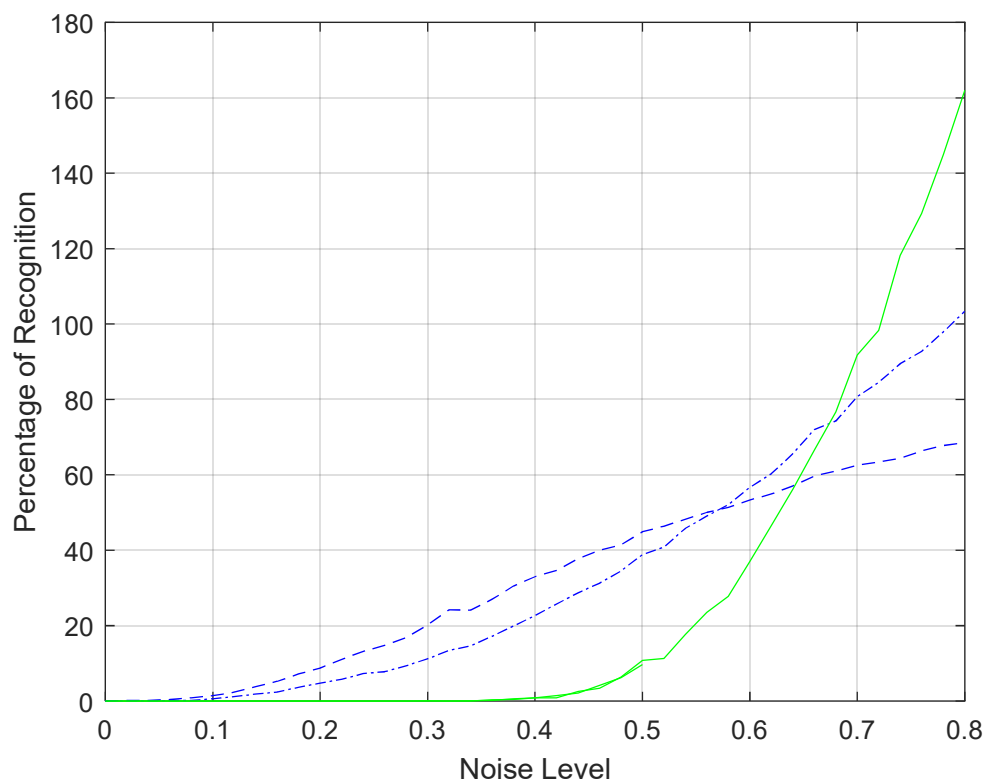


对比三次附加动量法的图（第三张附加动量因子为 0.9 的图在最后），可以发现误差大小和趋势没有大的变化，但是 BP 网络和单层感知器的性能转折点随着附加动量因子的增大而增大。

三. 实验分析和实验结论

1. 根据实验结果，对三种网络的抗噪性能进行对比分析：

- 首先，三种网络都可以达到含噪字符识别的目的，平均误差水平在 0.1 以下的时候，可以达到准确无误识别的目的。
- 随着误差的增加，使用误差数据训练的感知器网络要比使用同样范围误差训练的 BP 网络的精度稍微高，而 Hopfield 网络不仅不需要训练，语句简单，而且对误差的容忍程度要远高于感知器网络和 BP 网络。
- 我个人认为 BP 网络的性能肯定是要比感知器网络好的，一方面这次实验中感知器网络并没有明显优于 BP 网络，另一方面这次训练的数据像素很少才 35 个，并且灰度设置也只有 1 和 0，于是出现了这种意外，这可能是处于两种网络性能边界的一种状态，但是我相信一旦数据多而复杂肯定是 BP 网络占据优势。
- Hopfield 网络的误差曲线斜率越靠近 0.5 越大，说明它的增长会越来越快，所以说 hopfield 网络并不是完美的，在误差增长的时候它的错误率会急剧增加：



看上图，确实和预测的相同，hopfield 网络确实是误差大的时候表现质量迅速下滑，而一个意外是误差大的时候 BP 网络表现会迅速好于感知器网络，注意到这里的感知器网络和 BP 网络都是使用的平均误差在 0-0.5 的数据进行训练的。

- 总结：单层感知器网络结构和迭代算法都很简单，训练时间短，BP 网络结构和算法比单层感知器稍微复杂，而在表现上，在训练数据误差范围内，由于训练的数据过于简单，单层感知器网络的表现要优于 BP 网络，但是误差大的时候自然而然 BP 网络还是由于单层感知器网络的。Hopfield 网络无需训练，并且容错范围宽，但是在错误大的时候容错率就会比其他两种网络低很多。

2. 根据实验结果，探究所选变量对实验结果都影响并进行分析。

分析附在实验结果图后面。

3. 实验心得体会:

能 copy 代码就不要自己写, 花很多时间也容易出错, 遇见好的代码尽量分类整理好存到固态, 熟练使用 matlab 绘图功能, 另外我感觉实验难度对于非计算机专业的同学有点偏大, 第二个实验大概花了六七十小时, 只有实验四最简单, 另外两个都要花个二三十小时。

收获了一个简单易懂, 性能还非常好的 hopfield 网络, 寒假打算拿来搞事情。