

© 说明：成作业可以使用你所熟悉的编程语言和平台，比如 C，C++、MATLAB、Python等。作业链接。

- 人工神经网络（2024年秋季）：第二次作业：

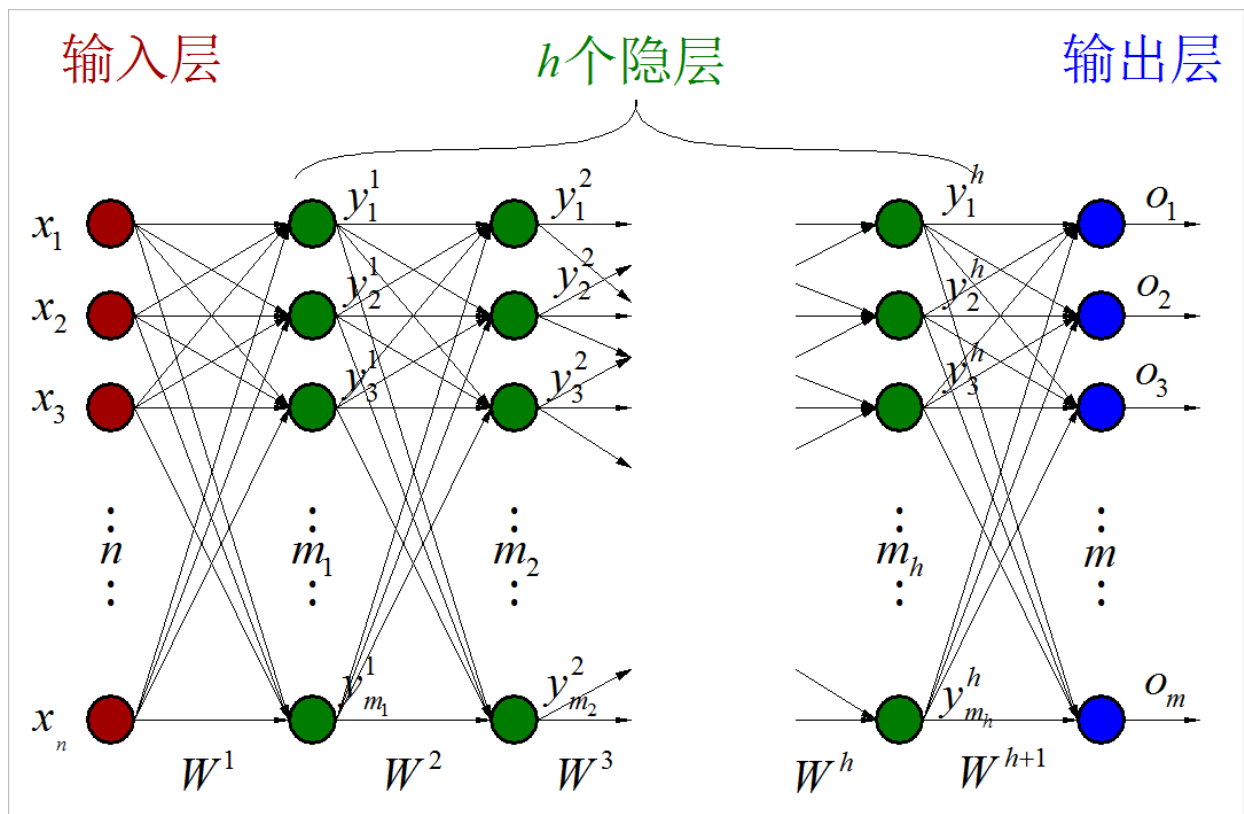
<https://zhuoqing.blog.csdn.net/article/details/142909470>

01 分类问题

一、基本BP算法网络

下面是分层前馈网络的示意图，分层前馈网络的**随机梯度下降算法**（BP算法）中，第 n 层的权系数 w_{ij}^n 的调整公式可以表述为：

$$\Delta w_{ij}^n = \eta \left(\sum_{k=1}^{m_n} \delta_k^{n+1} w_{jk}^{n+1} \right) y'^n \cdot y_i^{n-1}$$



▲ 图1.1.1 具有h+1层的BP网络结构示意图

1、作业基本要求

(1) 请写出如下图**结构1**网络结构算法中，每个神经元权系数的更新公式 Δw_{nm} 。神经元传递函数可以选择：sigmoid 或者 tanh（双曲正切），同学们自行选择。

$$f_1(x) = \frac{1}{1 + e^{-x}}; \quad f_2(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

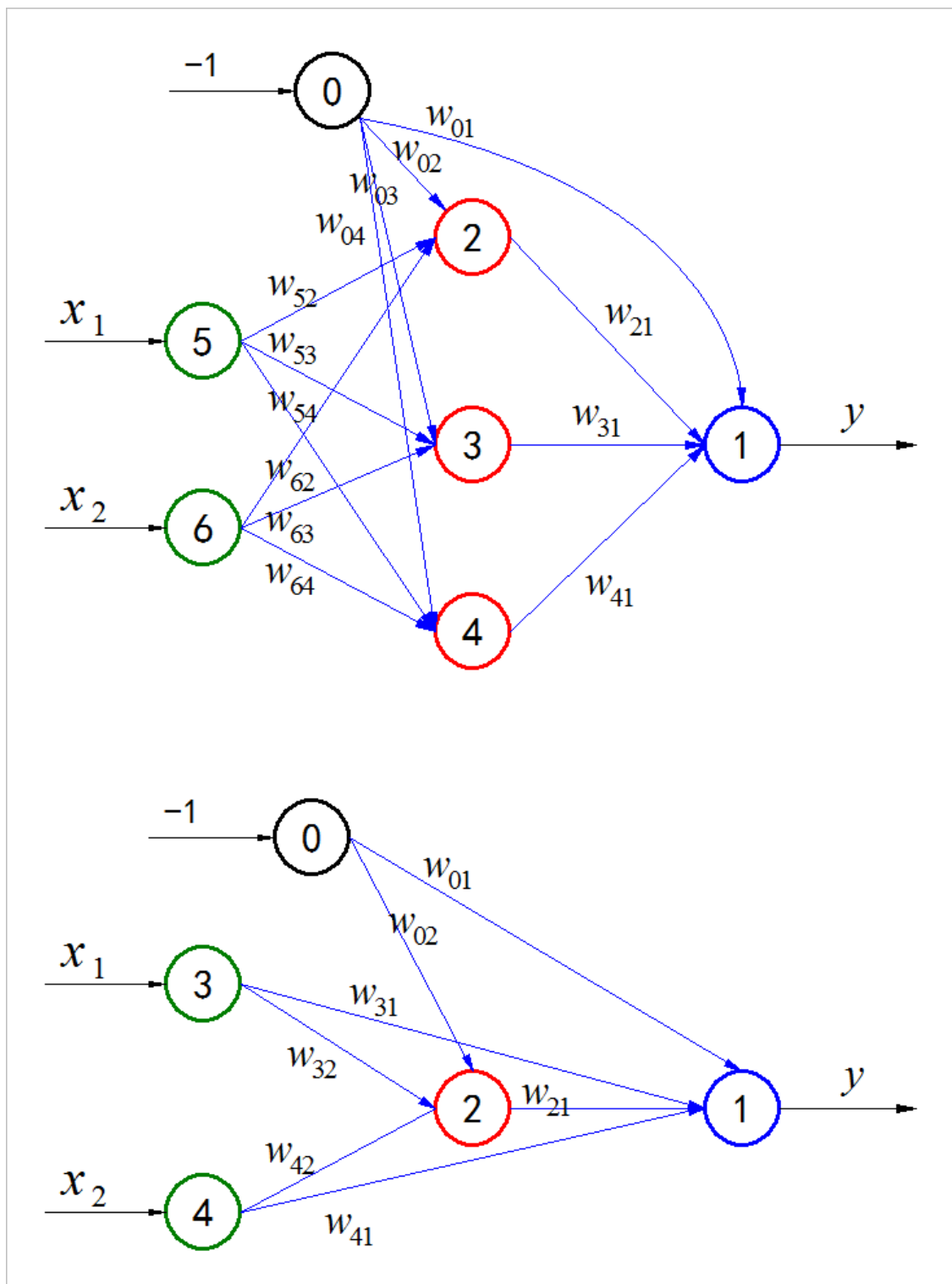
(2) 编写程序，实现**结构1**网络算法，求解 XOR 问题。

注：XOR问题中，四个样本对应的输入分别为

$$\bar{x}_{1,2,3,4} = \{[1, 1], [1, -1], [-1, 1], [-1, -1]\}$$

对应的期望输出分别为：

$$y_{1,2,3,4} = \{-1, 1, 1, -1\}$$



▲ 图 两种结构的神经网络

2、作业选做要求

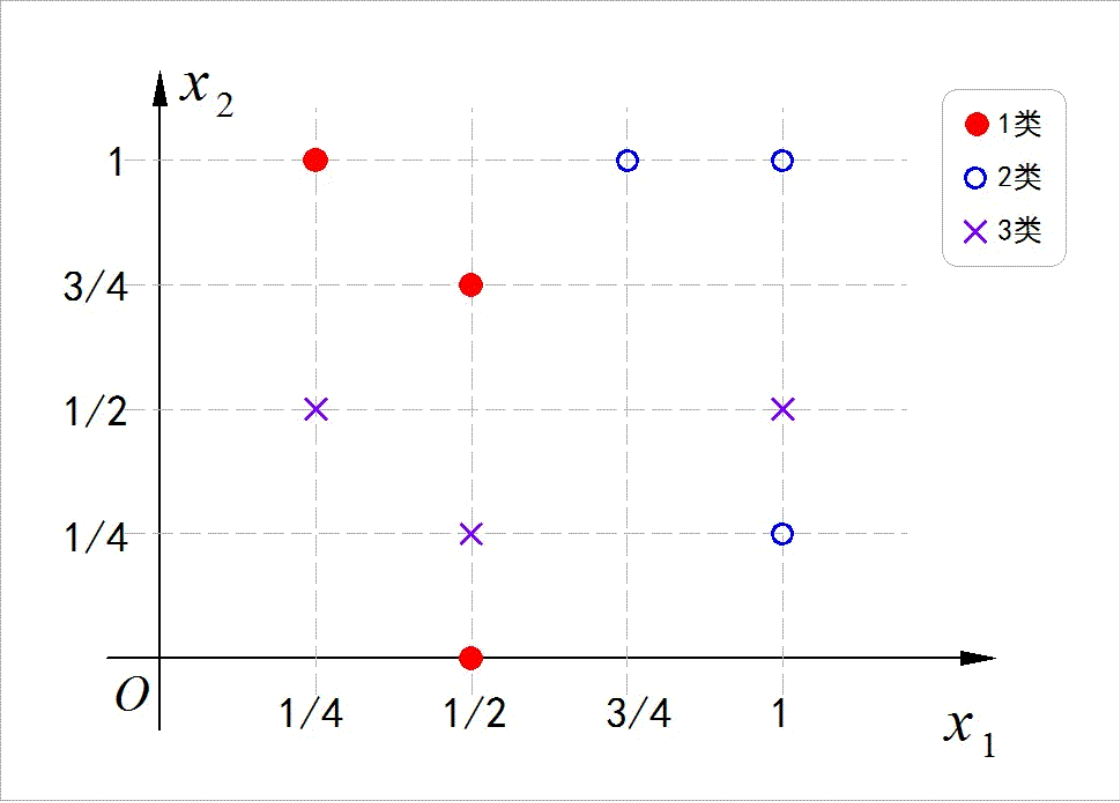
- (1) 请写出 **结构2** 网络中每个权系数的更新公式；
- (2) 编写程序实现**结构2**，求解 XOR 问题。

□ 参考答案：<https://zhuoqing.blog.csdn.net/article/details/109734950>

二、分类问题（三类）

1、分类问题

设计一个前馈分层神经网络，分别使用 **BP** 算法 和 **RBF** 算法，对于下图中平面坐标系中三类样本进行分类。每一类样本个数都是三个，它们的坐标 (x_1, x_2) 在 $[0, 1]$ 之间的五个离散值分布。



▲ 图1.1.1 三类样本处在坐标系中的位置

下面表格给出了三类样本五组随机分布的位置。可以任意选择一组分布作为训练样本训练网络。

【表1-1 五组三类分类坐标位置】

类别	位置1	位置2	位置3
类别1	(0.75;1.0)	(0.5;0.75)	(0.25;0.0)
类别2	(0.5;0.0)	(0.0;0.0)	(1.0;0.75)
类别3	(1.0;1.0)	(0.5;0.25)	(0.75;0.5)
类别1	(1.0;0.75)	(1.0;0.5)	(0.25;0.5)
类别2	(0.0;0.0)	(0.0;0.75)	(0.75;0.0)

类别	位置1	位置2	位置3
类别3	(1.0;1.0)	(0.75;1.0)	(0.25;0.25)
类别1	(0.25;0.5)	(0.0;0.25)	(0.75;0.0)
类别2	(0.0;0.0)	(0.75;0.5)	(1.0;0.5)
类别3	(0.75;0.75)	(1.0;0.25)	(0.0;0.75)
类别1	(0.0;0.0)	(0.25;0.5)	(0.75;0.0)
类别2	(0.75;0.5)	(0.25;0.25)	(0.5;0.25)
类别3	(1.0;0.25)	(1.0;1.0)	(0.5;0.75)
类别1	(0.75;0.5)	(0.0;0.5)	(0.5;0.75)
类别2	(0.0;0.75)	(1.0;0.5)	(0.75;1.0)
类别3	(0.0;0.0)	(0.75;0.0)	(0.75;0.75)

2、作业基本要求

(1) 使用单隐层BP网络进行分类

- 绘制出网络结构，并给出算法流程描述；
- 讨论不同隐层节点个数对于分类结果影响，并给出解决该分类问题最少隐层节点个数；
- 对每个样本增加噪声，讨论所训练网络的泛化能力。对样本增加噪声就是在两个坐标 (x_1, x_2) 上增加随机数，如下所示；

$$(x_1 + \delta_1, x_2 + \delta_2), \quad \delta_1, \delta_2 \in rand(-0.25, 0.25)$$

- 在网络训练过程中，请注意样本坐标进行归一化处理，即将坐标 x_1, x_2 经过变换，使其数值分布在 $(-1, 1)$ 的对称范围内，对比网络训练效果；

(2) 使用RBF网络进行分类

- 使用正规化RBF网络求解，给出网络参数与仿真结果；
- 使用广义RBF网络求解，并给出隐层节点个数分别为 2,3,4时对应的分类结构；

3、作业选做要求

(1) 训练数据扩充

选择一种网络（BF，RBF，SVM），对比以下两种数据训练后的网络泛化性能：

(1) 仅仅使用没有增加噪声的九个样本进行训练；

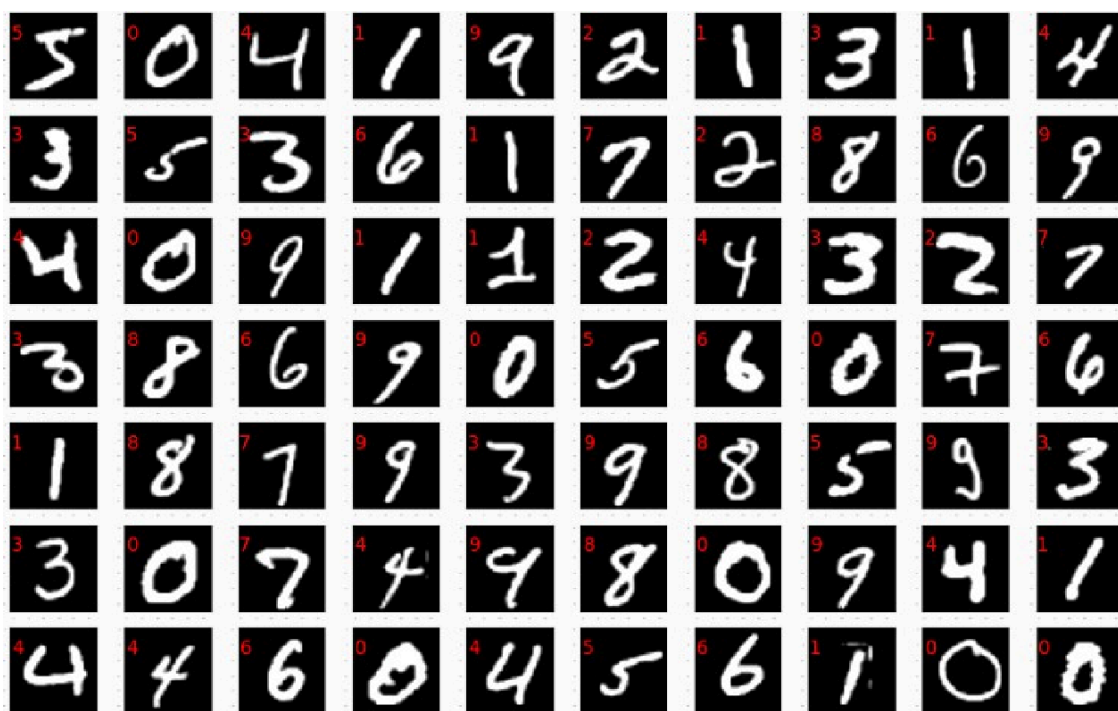
(2) 对样本增加随机噪声，扩充到 180 个样本进行训练。扩充后的样本个数可以自己定义；

□ 参考答案：<https://zhuoqing.blog.csdn.net/article/details/109745214>

三、MNIST分类

1、MNIST数据集合

设计一个BP网络完成MNIST手写体识别。



▲ 图1.2.1 MNIST 数据集合示例

□ MNIST数据库下载：<http://yann.lecun.com/exdb/mnist>

2、作业基本要求

(1) 直接图片输入：直接构造784个节点输入，10个节点输出，自行选择中间隐层数量和节点个数，使用50000数据进行训练，10000个测试样本测试。给出网络结构设计，训练方法以及识别结果。

3、作业选做要求

(1) 人工提取特征：参照课件上[3-4-4]“手写体识别”中介绍的方法，利用数字的边缘方向图与灰度图的降采样作为特征，使用小的神经网络进行训练。训练样本选择5000，测试样本选择2000。

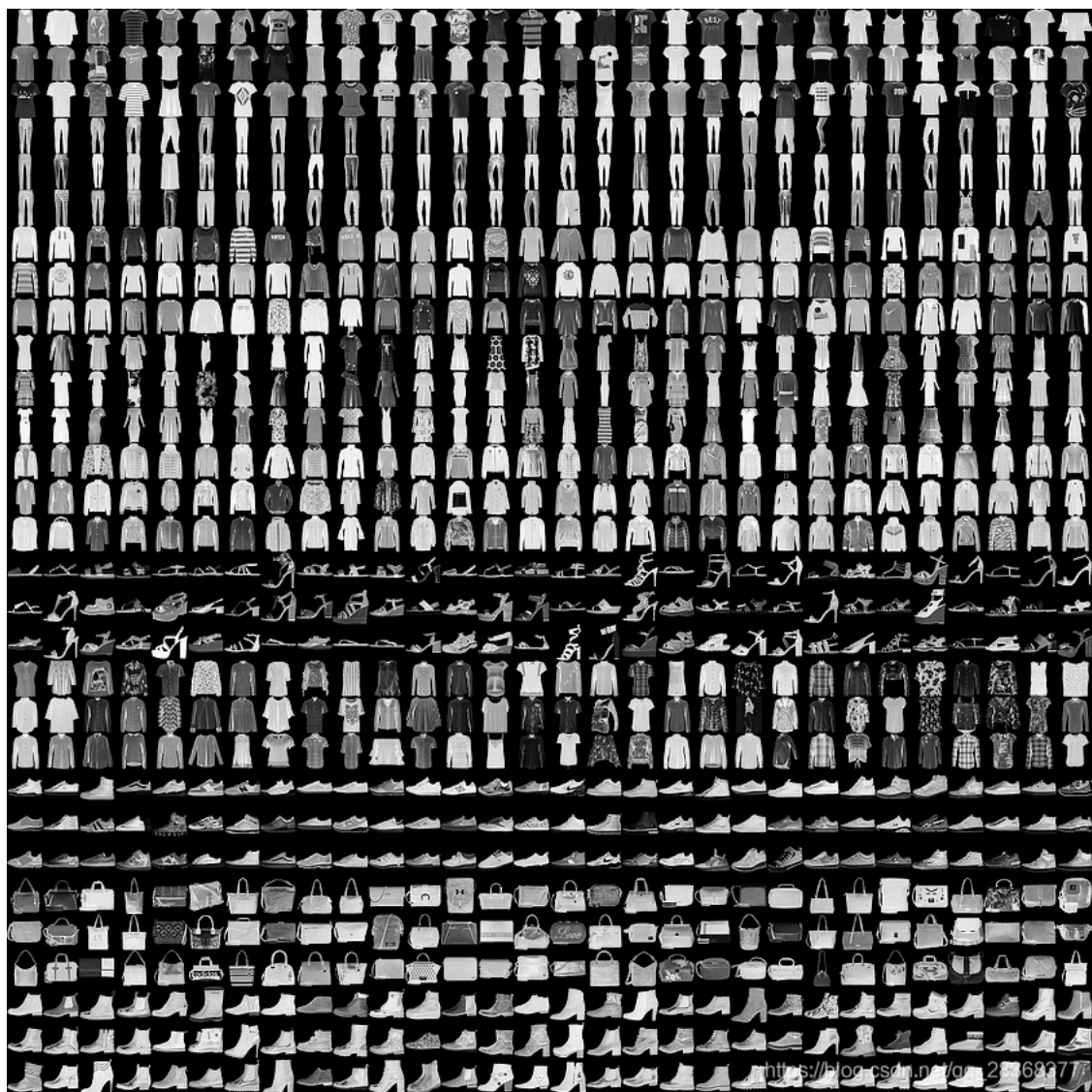
(2) 对比和讨论人工提取特征与直接图片输入两种方法的优缺点。

三、Fashion MNIST分类（选做）

1、Fashion MNIST数据集

设计一个BP网络完成Fashion MNIST手写体识别。 Fashion-MNIST是一组28x28灰度的衣服图像。 它比MNIST更复杂，因此可以更好地表示神经网络的实际性能。

Fashion MNIST/服饰数据集包含70000张灰度图像，其中包含60,000个示例的训练集和10,000个示例的测试集，每个示例都是一个28x28灰度图像，分为以下几类：



▲ 图1.4.1 Fashion MNIST 数据集

02 回归问题

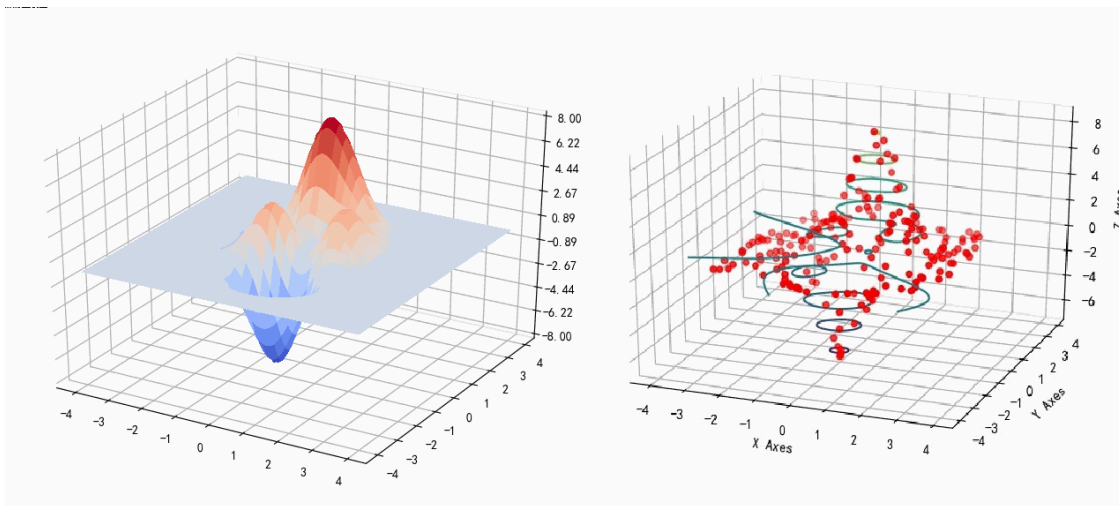
一、Peaks函数逼近

1、设计要求

使用神经网络逼近 MATLAB中的 peaks 二维函数。下面是 peaks 函数的数学表达式。

$$f(x, y) = 3(1-x)^2 e^{-[x^2+(y+1)^2]} - 10\left(\frac{x}{5} - x^3 - y^5\right) - \frac{1}{3}e^{[(x+1)^2+y^2]}$$

下面是 Peaks 函数的图像以及对应的网络样本采样点的分布。



▲ 图2.1.1 Peaks 函数的图像（左）以及训练样本采样点（右）

2、作业基本要求

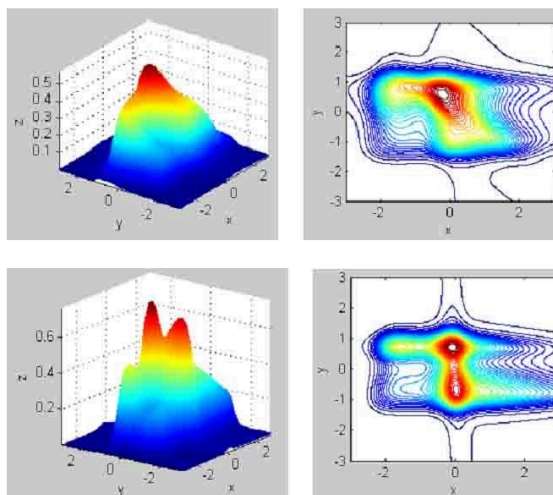
分别设计 BF，RBF 网络完成对 Peaks 函数的逼近。

(1) 样本数据采样区域在区域内 $[-4,4] \times [-4,4]$ 随机采样。样本采样的个数自行确定。

(2) 分别使用BP网络，RBF网络实现上述函数逼近；

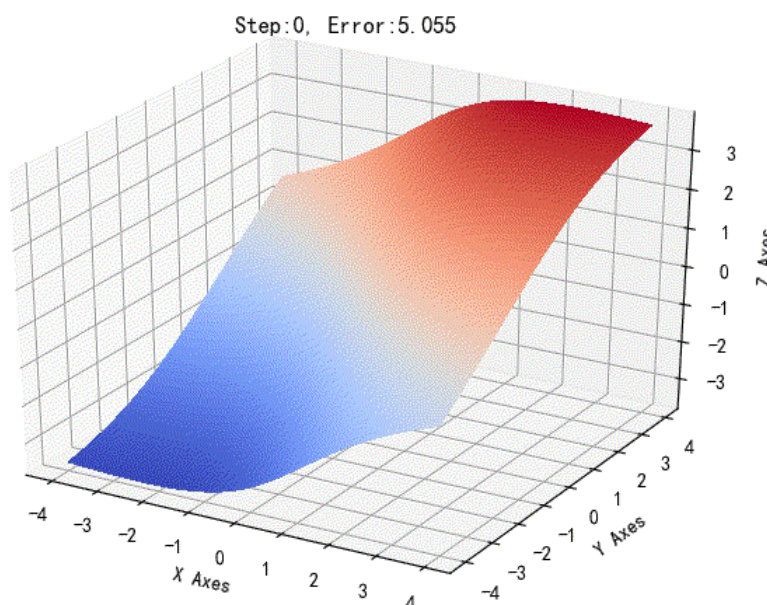
(3) 在BP网络中，讨论训练样本个数与训练样本采样的分布对函数逼近效果的影响。讨论参见课件【3-4-2：函数逼近问题的讨论】。

- 采用150个样本，
- 学习速率0.1，
- 动量因子0.4，
- 隐节点7个，
- 训练步数15000步
- 采用200个样本，
- 200个带信息的样本：
- 一半随机取，
- 一半来自特征区域



▲ 图2.1.2 训练样本采集是否集中在函数变化剧烈的区域

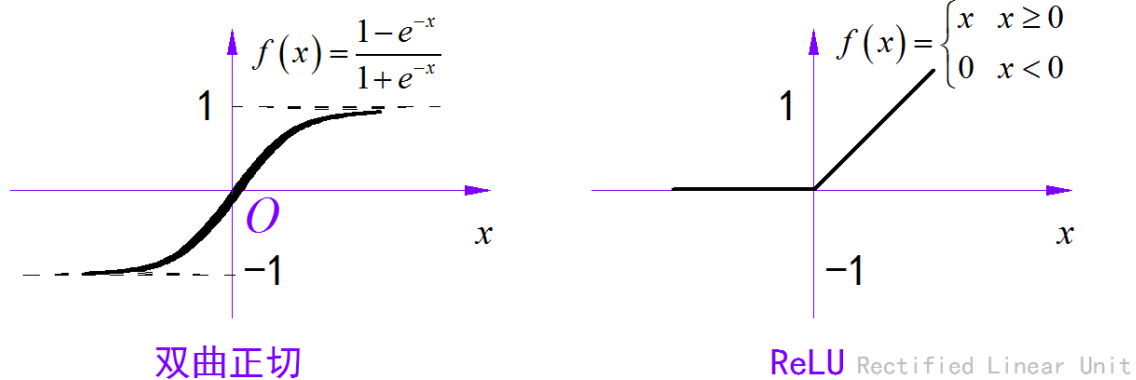
(4) 在RBF网络中，讨论隐层神经元尺度参数对于函数逼近的影响。



▲ 图2.1.3 随着训练次数增加对应的函数逼近曲面

3、作业选做要求

(1) 在BP网络中，请对比中间隐层传递函数在使用一下两种函数时对于结果的影响：



- 参考答案: 2020人工神经网络第一次作业-参考答案第五部分

一、ASCII字母压缩

2、作业基本要求

[illegible]

- (2) 讨论网络隐层节点个数与恢复数据误差之间的关系。
- (3) 给出隐层节点在15个时, 26个字母压缩恢复后的数据图像。

□ 参考答案：<https://zhuoqing.blog.csdn.net/article/details/109830470>

二、图像压缩（选做）

☐ 这个题目是选做内容。

1、设计要求

使用神经网络对于Lena图像进行压缩。对于隐层节点数目在16~256之间选择几个数目，给出图像压缩的效果以及图像压缩MSE。

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^N (I_i^{org} - I_i^{re})^2$$

其中 I_i^{org} 是原始图像灰度, I_i^{re} 是恢复图像的灰度值。



▲ 图3.2.1 灰白Lena测试图片

注：Lena的灰度图像可以从网络上寻找并下载。也可以选择其它的图像进行实验。

- 参考答案： **2020人工神经网络第一次作业-参考答案第八部分**

注：Lena的灰度图像可以从网络上寻找并下载。也可以选择其它的图像进行实验。
下面给出了Lena下载链接：

- **BMP-GRAY:Lean512** :
<https://www.cosy.sbg.ac.at/~pmeerw/Watermarking/lena.html>
- **MATLAB:Lean512.mat(262kb)** :
<https://www.ece.rice.edu/~wakin/images/lena512.mat>

※ 附件 ※

一、三类问题数据生成

下面的 Python 程序完成三类分类问题中，三类样本随机生成的算法，同时生成不同样本集合对应位置动图。

```
from headm import *

pointdim = [30, 31, 32, 35, 34, 33, 36, 37, 38]
xyc = XYCoor(2, 3, 7, 8, 1, 1)
plotbox = 41

plotgif = PlotGIF()

xrange = list(arange(5)/4)*4
yrange = [xx for x in zip(*[list(arange(5)/4)]*4) for xx in x]
xyrange = [(x,y) for x,y in zip(xrange, yrange)]

printf('类别 位置1 位置2 位置3')
for i in range(5):
    random.shuffle(xyrange)

    for j in range(3):
        printf('类别%d'%(j+1),
               str(xyrange[j*3+0]).replace(' ', '').replace(',', ';') + ' ',
               str(xyrange[j*3+1]).replace(' ', '').replace(',', ';') + ' ',
               str(xyrange[j*3+2]).replace(' ', '').replace(',', ';') + ' ')

    printf('- - - -')

    for j in range(len(pointdim)):
        xd,yd = xyrange[j]

        x,y = xyc.XY2Pos(xd,yd)
        tsplocatedop(pointdim[j], x,y)

    tsprv()
    time.sleep(.05)
    plotgif.appendbox(plotbox)

plotgif.save(r'd:\temp\gif1.gif', 500)

printf('\a')
```

二、函数显示

下面 Python 程序绘制 Peaks 函数3D显示。


```

from headm import *

from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
from matplotlib import cm
from matplotlib.ticker import LinearLocator, FormatStrFormatter
import numpy as np

fig = plt.figure()
ax = fig.gca(projection='3d')

X = np.arange(-4, 4, 0.05)
Y = np.arange(-4, 4, 0.05)
X, Y = np.meshgrid(X, Y)

def fxy(x,y):
    e1 = exp(-(x**2+(y+1)**2))
    e2 = exp(-(x**2+y**2))
    e3 = exp(-((x+1)**2+y**2))

    fvalue = 3*(1-x)**2*e1 - 10*(x/5-x**3-y**5)*e2 - 1/3*e3
    return fvalue

Z = fxy(X,Y)

surf = ax.plot_surface(X, Y, Z, cmap=cm.coolwarm,
                      linewidth=0, antialiased=False)

ax.set_zlim(-8, 8)
ax.zaxis.set_major_locator(LinearLocator(10))
ax.zaxis.set_major_formatter(FormatStrFormatter('%.02f'))

fig.colorbar(surf, shrink=0.5, aspect=5)

plt.show()

```

■ 相关文献链接：

- [2023秋季学期人工神经网络](#)
- [2020人工神经网络第一次作业-解答第一部分](#)
- [2020人工神经网络第一次作业-参考答案第二部分](#)
- [MNIST数据库下载](#)

- [2020人工神经网络第一次作业-参考答案第五部分-CSDN博客](#)
- [2020人工神经网络第一次作业-参考答案第七部分](#)
- [2020人工神经网络第一次作业-参考答案第八部分-CSDN博客](#)
- [BMP-GRAY:Lean512](#)
- [MATLAB:Lean512.mat\(262kb\)](#)

● 相关图表链接：

- [图1.1.1 具有h+1层的BP网络结构示意图](#)
- [图1.1.2 两种结构的神经网络](#)
- [图1.1.1 三类样本处在坐标系中的位置](#)
- [表1-1 五组三类分类坐标位置](#)
- [图1.2.1 MNIST 数据集合示例](#)
- [图2.1.1 Peaks 函数的图像（左）以及训练样本采样点（右）](#)
- [图2.1.2 训练样本采集是否集中在函数变化剧烈的区域](#)
- [图2.1.3 随着训练次数增加对应的函数逼近曲面](#)
- [图2.1.3 两种不同的神经元传递函数](#)
- [图3.1.1 字符点阵信息与压缩网络](#)
- [图3.2.1 灰白Lena测试图片](#)