学习小组 2 机器学习期中大作业

小组成员					
王锦宏	吴泽辉	陶少聪	王海东	叶泽林	陈耀顺
19351125	19351146	19351119	19351124	19351163	18364013

（中山大学智能工程学院，广东 深圳，518107）

题目：(神经网络)某型设备从投入使用到最终报废的整个寿命周期内，设备的健康状态大致可分为 3 个阶段：“初期磨合”、“稳定运行”和“快速老化”。为了对该设备的健康状态进行有效的监控，布置多种传感器来采集设备的运行状态。对传感器数据进行初步的信号处理，可获得一组 70 维的数据特征。通过模型训练学习这 70 维的数据特征与设备 3 个阶段的健康状态的关系，即可通过传感器数据的获取与分析来实现对设备所处的健康状态的监测与判断。

1 十折交叉验证

交叉验证法，就是将一个数据集分为 K 份，然后取其中一份作为测试集，剩余 K-1 份作为训练集。然后，取另一份作为测试集，其余 K-1 份作为训练集。如此循环，直到每一份都做过测试集为止。

本题当中采取的是 10 折交叉验证，即取 k=10 时对数据进行处理。每次试验都会得出相应的正确率（或差错率）。10 次的结果的正确率（或差错率）的平均值作为对算法精度的估计。一般还需要进行多次 10 折交叉验证（例如 10 次 10 折交叉验证），并且采取不同的划分数据集的方式，减小因样本划分不同而产生的误差。再求其均值，作为对算法准确性的估计。

本题中使用的十折交叉验证方法使用 sklearn.model_selection 库中的 KFold 方法实现，这是一种封装非常完善的十折划分方法，能够返回不重复的十折划分 index。为了保证十折效果的特异性，我们启用了随机划分的方法，使得十折训练数据和测试数据都能够取到原数据各层的训练结果且不重复，并定义 random_state=23 以便在不同机器上观测到相同的十折验证结果。

同时十折交叉验证法作为一种对数据处理结果的评估算法，也可以用于对不同模型拟合结果的一种选择。

2 RBF 神经网络

在 RBF 神经网络中，用 RBF 作为隐单元的“基”构成隐含层空间，这样就可以将输入矢量直接映射到隐空间，隐层把向量从低维度映射到高维度，这样低维度线性不可分的情况到高维度就可以变得线性可分了。

其中，隐层的各个神经元，也就是中心点，通过 k-means 聚类确定。当隐层的神经元确定后，隐含层神经元的输出便是已知的，这样的神经网络的连接权就可以通过求解线性方程组来确定，就可以基于矩阵伪逆思想解出各连结权。

RBF 神经网络示意图如图 1 所示。

特别指出，数据集$D = \{(x_1, y_1), \cdots, (x_p, y_p)\}$，其中$x_i = (x_1; \cdots; x_{70})$，$y_i = (y_1; y_2; y_3)$。

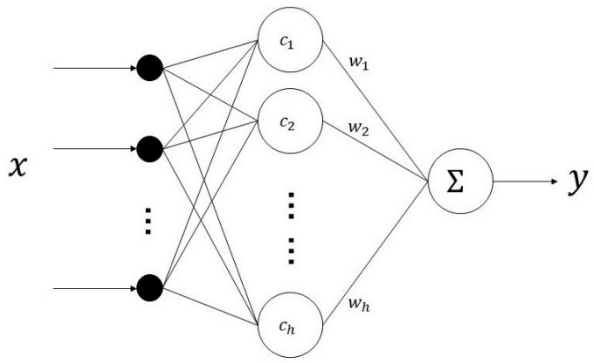

图 1 RBF 神经网络示意图

2.1 隐层

选定隐层神经元的个数，也就是聚类的中心点。直接选取数据前 h 个作为初始中心点。

2.2 聚类

2.2.1 针对每个样本点，找到距离其最近的中心点，距离同一中心点最近的点为一个类，这样完成了一次聚类。

$$min\{|x_i - c_j|\} \quad \begin{matrix} i = & 1,\cdots,p \\ j = & 1,\cdots,h \end{matrix}$$

2.2.2 计算每一个聚类的中心点，如果所有类的中心点与上次相同，则分类完成；否则，更新中心点返回步骤 2.2.1。

2.3 激活函数

径向基神经网络的激活函数，可表示为

$$r_{ij} = exp\left(-\frac{1}{2\sigma^2}\|x_i - c_j\|^2\right)$$

其中r_{ij}可以组成受激励矩阵Γ。

2.4 高斯核函数的径向基

对于高斯核函数的径向基，方差由如下公式求解：

$$\sigma_i = \frac{c_{max}^2}{\sqrt{2h}} \quad i = 1,2,\cdots,h$$

其中，c_{max}为所选取中心点之间的最大距离。

方差参数决定了隐神经元对外部输入信号的响应范围，每个隐神经元的中心点位置和响应宽度范围可以是不同的，分别负责各自的局部映射动作。

当输入信号靠近某个隐神经元中心时，则该神经元被激活，产生较大输出；当输入信号远离这个中心时，则该神经元的输出趋于零。

2.5 网络输出

其中x_p径向基神经网络的结构可得到网络的输出为：

$$\hat{y}_{pj} = \sum_{i=1}^{h} exp\left(-\frac{1}{2\sigma^2}\|x_p - c_i\|^2\right)w_{ij} \quad j = 1,2,3$$

2.6 连接权值

我们可以基于矩阵伪逆思想直接确定 RBF 网络连接的权值w_i。

径向基函数只对输入信号产生局部响应，隐含层神经元的输出在输出层进行进一步的线性加权求和，实现从输入空间到输出空间的映射，从而使整个网络达习j分类或函数逼近的目的。

$$W = \Gamma^+ Y^T$$

本题的输出有三个特征值，所以W是w_i组成的 h 行 3 列的权值矩阵。

受激励矩阵为

$$\Gamma \in R^{p\times h} = [r_{ij}] \quad \begin{matrix} i \in & \{1,\cdots,g\} \\ j \in & \{1,\cdots,h\} \end{matrix}$$

其中，Γ^+是激励矩阵的伪逆矩阵，$Y \in R^{3\times p}$是训练样本对的输出向量。

2.7 输出值y的归一化

输入测试集测试时，对y进行归一化处理：

如果

$$y_i = \max\{y_1, y_2, y_3\}$$

那么

$$y_i = 1$$

否则

$$y_i = 0$$

因此(1,0,0)对应题目标记 0，(0,1,0) 对应题目标记 1，(0,0,1) 对应题目标记 2，由此判断为第几类。

3 训练算法

3.1 求解方差

高斯函数中的方差由下列公式直接确定

$$\sigma_i = \frac{c_{max}^2}{\sqrt{2h}} \quad i = 1,2,\cdots,h$$

其中，c_{max}为所选取中心点之间的最大距离，h为中心点个数。

3.2 求解连接权值

因为隐层与输出层之间为线性关系，因此隐层与输出层之间的权值可直接通过矩阵运算求解

$$W = \Gamma^+ Y^T$$

本题的输出有三个特征值，所以W是w_i组成的 h 行 3 列的权值矩阵。

受激励矩阵为

$$\Gamma \in R^{p\times h} = [r_{ij}] \quad \begin{matrix} i \in & \{1,\cdots,g\} \\ j \in & \{1,\cdots,h\} \end{matrix}$$

其中，Γ^+是激励矩阵的伪逆矩阵，$Y \in R^{3\times p}$是训练样本对的输出向量。

4 模型训练结果

4.1 十折交叉训练结果

注意：我组的数据在十折运算时可以选择被随机打散，或者按顺序被十折分割。我们对两种分割方式均进行了测试，结果如下：

按顺序进行十折交叉运算		
十折训练组	正确/测试总数	正确率
1	71/72	0.986111
2	71/72	0.986111
3	70/72	0.972222
4	70/72	0.972222
5	71/72	0.986111
6	71/72	0.986111
7	70/72	0.972222
8	71/72	0.986111
9	71/72	0.986111
10	70/72	0.972222
十折平均准确率：0.977778		

我们将十折数据以 23 为随机数种子随机分割，获得了如下的结果：

以 23 为随机数种子 打乱分割，进行十折交叉运算		
十折训练组	正确/测试总数	正确率
1	69/72	0.958333
2	69/72	0.958333
3	71/72	0.986111
4	70/72	0.972222
5	72/72	1.000000
6	71/72	0.986111
7	71/72	0.986111
8	70/72	0.972222
9	71/72	0.986111
10	71/72	0.986111
十折平均准确率：0.979167		

4.2 对自身的拟合结果

RBF 神经网络对于已学习的无冲突内容可以做到正确率 100%。在我组使用全数据训练并重新预测模型时，正确率为 100%。

对自身进行拟合		
训练组	正确/测试总数	正确率
全数据自身	720/720	1.00

4.3 与前馈神经网络结果比较

在本题中我们还尝试了双隐层前馈神经网络，其中第一个和第二个隐层都包含 6 个神经元，输出为 3 个神经元，分别对应题目的三种状态。

而前馈神经网络的相关结构与算法在上一次作业的报告中已经详细写出，这里不再赘述。

最终，我们使用十折交叉验证来测试前馈神经网络模型，结果记录在下表。

按顺序进行十折交叉运算		
十折训练组	正确/测试总数	正确率
1	69/72	0.958333
2	72/72	1.000000
3	68/72	0.944444
4	71/72	0.986111
5	69/72	0.958333
6	70/72	0.972222
7	70/72	0.972222
8	70/72	0.972222
9	71/72	0.986111
10	72/72	1.000000
平均值	70.2/72	0.975000

相对于 BP 神经网络，RBF 网络在十折验证上的结果明显优于 BP 网络，同时使用 Python 建成的 RBF 网络由于没有前向传播，隐藏层到输出层的参数使用正则化方法直接计算，因此效率显著快于 BP 网络。在我组的测试环境中，**RBF 网络比 BP 网络的训练速度快了将近 1000 倍(优化梯度下降算法后约快 2 倍)**。

5 总结模型训练过程中的收获

5.1 模型调试

对于 RBF 神经网络来说，最重要的就是隐层神经元的确定，而连结权的在隐层神经元确定后由矩阵伪逆可以直接确定。

在开始时用一部分训练数据的点作为隐层神经元，没有使用 K-means 聚类的方法，效果不太理想。故为了得到更好的效果，便使用了 K-means 聚类的方法来确定隐层神经元，效果得到了很大的提升。

5.2 不同网络比较

对于本次的题目我们小组共尝试了两种神经网络模型，一个是 BP 神经网络，一个是 RBF 神经网络。

RBF 神经网络经过十折交叉验证后准确率略

高于 BP 神经网络，同时，RBF 网络运行速度远快于 BP 网络，故对本次的题目我们小组最终选用 RBF 神经网络模型。