神经网络的目标是网络的期望输出与实际输出结果的误差函数值最小, 因此本文构造的适应度函数为:

$$f = \frac{1}{E(w,b)} * 10^{-3} \tag{2}$$

1.4 选择算子 (selection operator)

选择策略是对当前群体不同适应度个体进行优胜劣汰的过程, 本文采用轮盘赌和最佳保留相结合的方法选择个体.

1.5 交叉算子 (crossover operator)

交叉算子是用根据选择操作得到的两个染色体个体, 以一定的概率 P_c 按照一定的方式互换一些基因, 从而得到子代染色体的过程. 交叉概率越大, 子代染色体更新的越快, 然而就更容易破坏优良个体. 交叉概率越小, 算法的收敛速度越慢, 通常 P_c=0.5~1.0. 为了尽量不破坏适应度值高的染色体个体, 同时保证群体的多样性, 本文采用自适应的交叉率[11]公式如下:

$$\begin{cases} P_{c1} - \dfrac{(P_{c1} - P_{c2}) / \left(f_{\max} - f_{\text{avg}}\right)}{f_{\max} - f_{\text{avg}}}, f' > f_{\text{avg}}; \\ P_{c1}, \qquad\qquad\qquad\qquad f' < f_{\text{avg}} \end{cases} \tag{3}$$

其中, f_{avg}为平均适应度, $f_{\max}$为群体中个体最大的适应度, f'为交换的两个个体中适应度值大的个体.

本文在进行交叉操作时, 把父代染色体对应基因位的染色体互换, 即控制基因位与对应的控制基因位互换, 权重系数基因位与对应的权重系数基因位互换, 阀值基因位与阀值基因位互换.

1.6 变异算子 (mutation operator)

变异操作是通过将染色体个体的编码基因中某些位用其它等位基因代替, 从而产生新的个体, 通常 P_m=0.001~0.05. 本文采用自适应的变异率, 公式如下:

$$\begin{cases} P_{m1} - \dfrac{(P_{m1} - P_{m2}) / \left(f_{\max} - f_{\text{avg}}\right)}{f_{\max} - f_{\text{avg}}}, f' > f_{\text{avg}}; \\ P_{m1}, \qquad\qquad\qquad\qquad f' < f_{\text{avg}} \end{cases} \tag{4}$$

其中, f_{avg}、$f_{\max}$、f'见上文.

以P_m的概率对交叉操作以后的染色体进行变异, 变异算子如下:

$$X_j^{t+1} = X_j^t + c_j \tag{5}$$

其中, X_j^t是变异操作前的个体, X_j^{t+1}是变异操作后的个体, c_j是随机数.

1.7 终止条件

本文设计的算法终止条件如下:

当算法运行到预先设定的最大的进化代数 K_0 时, 就终止算法, 把得到的结果输出.

1.8 算法框架

遗传神经网络算法具体步骤如下:

(1) 确定解空间[12], 对解空间进行编码, 每串编码代表解空间的一个解.

(2) 在编码的解空间中, 随机生成一个初始群体 (不要求一定是可行解).

(3) 对群体中的每一个个体进行适应度评价.

(4) 根据个体的适应度, 对群体中的个体进行选择、交叉、变异遗传操作.

(5) 生成新一代群体.

(6) 反复进行 (3)、(4)、(5), 每进行一次, 群体进化一代, 直至进化了 K_0 代 (K_0 为预置的进化代数).

(7) 从第 K_0 代群体中选择 S 可能具有全局性的进化解, 选择时要注意避免相似个体.

(8) 分别以这些进化解为初始解, 用神经网络求解.

(9) 比较 S 个由神经网络求得的解, 从而获得问题的最优解, 然后输入检测样本进行预测.

遗传神经网络算法流程图如图 2 所示.

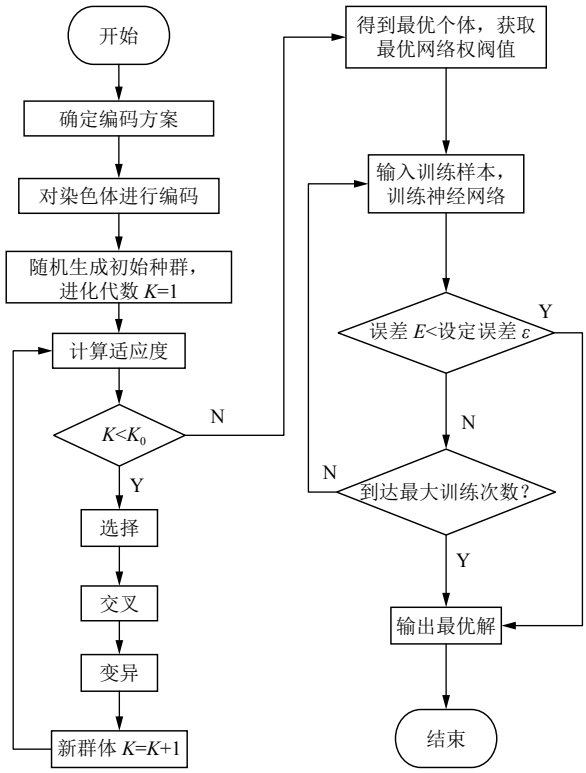

图 2 遗传神经网络算法流程图

2 GA-BP 算法的并行化

Hadoop 下的 MapReduce 是 Google 公司提出的用于处理海量数据的分布式计算模型, 其主要由两个阶段组成: Map 和 Reduce. Map 阶段主要负责将输入数据分为多个分片 (split), 并将每个 split 交给一个 Map Task, 最终转化为 key/value 数据结构. Reduce 阶段将 Map 阶段的结果进行归约处理, 输出最终结果. 降水数据规模较大, 利用 MapReuce 分布式计算的优势, 可以快速的完成对海量气象数据的处理.

2.1 遗传算法的并行化

GA 算法的并行思想: 首先将整个群体分为 m 个种群, 然后每一个种群在对应的 Map Task 上独立完成种群的初始化、选择、交叉、变异等操作, 达到收敛条件后, 将 Map 得到的最后个体传递给相对应的 Reduce Task, 然后将不同 Reduce Task 得到的个体适应度进行比较, 输出适应度值最大的个体.

Map 函数伪代码如下:

```
Input: key, value
Output: key, best individual set
Public void map (writeable key, value){
 InitPopulation();
 EvaluateFitness();
 Do{
  Selection operator();
  Crossover operator();
  Mutation Operator();
}while (K<=K0)
OutputBestIndividual();}
```

Reduce 函数伪代码如下:

```
Input: key, best individual set
Output: key, best individual
void reduce(individual, fitness)
{
 CompareIndividualFitness();
 Output();
}
```

遗传算法迭代完成后, 得到的结果就是 BP 神经网络的初始的权值和阀值.

2.2 BP 神经网络的并行化

神经网络算法的并行思想: 在 Map 阶段, 根据遗传算法输出的最优解得到神经网络的初始权值和阀值, 然后把训练样本转化为键值对作为输入, 对每个样本进行迭代运算, 计算误差, 并反向传播误差, 输出权值的改变量. 在 Reduce 阶段, 对 Map 输出的各个权值的改变量进行累加并求平均值, 对权值进行更新. 重复 Map 和 Reduce 阶段, 直到算法收敛.

Map 函数伪代码如下:

```
Input: key, value
Output: key, weight change
Public void map (writeable key, value){
 Do{
 //For each sample
  Error();
  WeightChange();
 }while (E<Ɛ)
 OutputWeightchange();}
```

Reduce 函数伪代码如下:

```
Input: key, weightchange
Output: key, mean weightchange
void reduce(key, weightchange){
 Do{
  SumWeightchange();
  MeanWeightchange();
 }while()
Output();}
```

3 实验结果及分析

3.1 实验环境

实验采用 Hadoop 分布式集群[13], 选取 1 台虚拟机作为 NameNode 节点和 JobTracker 服务节点, 负责管理分布式数据和分解任务的执行, 其它 7 台虚拟机作为 DataNode 和 TaskTracker 服务节点, 负责分布式存储和任务执行. 虚拟机各项配置及集群的配置信息分别如表 1、表 2 所示.

表 1 虚拟机配置信息表

名称	配置
CPU	Intel(R) Xeon(R) CPU E7- 4807
内存	16 G
硬盘	100 G
操作系统	CentOS 6.5
JDK	Jdk1.7.0_79
Hadoop	Hadoop-2.6.0
Hive	Hive-1.2.1

表 2 集群信息配置

主机名	IP	进程
Master	10.xxx.xxx.23	NameNode、DataNode、JobHistoryServer、NodeManager
Slave1	10.xxx.xxx.24	DataNode、NodeManager、ResourceManager、WebAppProxyServer
Slave2	10.xxx.xxx.25	SecondaryNameNode、DataNode、NodeManager
Slave3	10.xxx.xxx.26	DataNode、NodeManager
Slave4	10.xxx.xxx.27	DataNode、NodeManager
Slave5	10.xxx.xxx.28	DataNode、NodeManager
Slave6	10.xxx.xxx.29	DataNode、NodeManager
Slave7	10.xxx.xxx.30	DataNode、NodeManager

由表 2 我们可以看出 Hadoop 分布式集群在运行时需要一系列的后台程序, 主要有:

NameNode-负责管理文件系统的 Namespace.

DataNode-负责管理各个存储节点.

SecondaryNameNode-NameNode 的热备, 负责周期性的合并 Namespace image 和 Edit log.

ResourceManager-负责调度资源.

NodeManager-负责管理 slave 节点的资源.

3.2 实验数据及预处理

实验采用的数据来自于天津市地面气候资料日值数据集, 该数据集以天津市 13 个站 1951 年~2006 年各月的 A0、D、A 文件为数据源, 通过统计软件处理转换成日平均气压、日最高气压、日最低气压等气象要素日值资料, 如表 3 所示.

表 3 天津市地面气候资料日值数据集表

序号	要素	序号	要素
1	区站号	15	平均相对湿度
2	纬度	16	最小相对湿度
3	经度	17	20–8 时降水量
4	拔海高度	18	8–20 时降水量
5	年	19	20–20 时降水量
6	月	20	小型蒸发量
7	日	21	大型蒸发量
8	平均本站气压	22	平均风速
9	日最高本站气压	23	最大风速
10	日最低本站气压	24	最大风速的风向
11	平均气温	25	极大风速
12	日最高气温	26	极大风速的风向
13	日最低气温	27	日照时数
14	平均水汽压		

按照中国气象局的划分, 将 20-20 时降水量划分为 7 类, 如表 4 所示, 实验中不考虑 31XXX 降雪以及 32XXX 雾露霜天气.为使网络有良好的收敛性和映射能力, 消除原始数据形式不同所带来的不利, 通常的做法是归一化处理, 将原始目标、输入数据转换到区间内将数据归一化到[0, 1]之间. 归一化公式如下:

$$x_i = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \tag{6}$$

其中, x_i为输入数据, i 为数据序号, $x_{\max}$、$x_{\min}$为 x 中的最大值和最小值.

表 4 降水量等级分类标准

标识	降水量 (毫米)	类型
R_0	0	无雨
R_1	0.1~9.9	小雨
R_2	10.0~24.9	中雨
R_3	25.0~49.9	大雨
R_4	50.0~99.9	暴雨
R_5	100.0~249.9	大暴雨
R_6	>=250.0	特大暴雨

在实验中, 1951 年~2005 年的数据作为训练数据, 2006 年的数据作为检测数据. 经过数据预处理后 13 个台站共计 214 706 个样, 如表 5 所示.

表 5 站点数据表

站号	起止年月日	样本数
54 428	1957.1.1–2006.12.31	18 262
54 517	1958.1.1–2006.12.31	17 897
54 523	1959.1.1–2006.12.31	17 532
54 525	1959.3.1–2006.12.31	17 473
54 526	1955.1.1–2006.12.31	18 993
54 527	1951.1.1–2006.12.31	20 454
54 528	1958.1.1–2006.12.31	17 897
54 529	1964.1.1–2006.12.31	15 706
54 530	1974.1.1–2006.12.31	12 052
54 619	1959.1.1–2006.12.31	17 532
54 622	1974.1.1–2006.12.31	12 053
54 623	1951.1.1–2006.12.31	21 185
54 645	1986.1.1–2006.12.31	7670

总体样本降水等级分布如图 3 所示, 从图 3 中不难看出 13 个台站无雨的样本数最多, 达到了 80.32%, 特大暴雨的样本数最少, 约 1.85%.

3.3 降水因子选择

天津市地面气候资料日值数据集共包含了 27 个要素, 去除降水量还有 24 个要素, 而预测因子的选择很大程度上影响了预测的结果, 本文利用 Pearson[14]相关系数法来考察各要素对降雨量的影响力, 公式如下:

$$r = r_{xy}\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}} \quad (7)$$

其中, x 与 y 分别为 2 个变量的观测值.

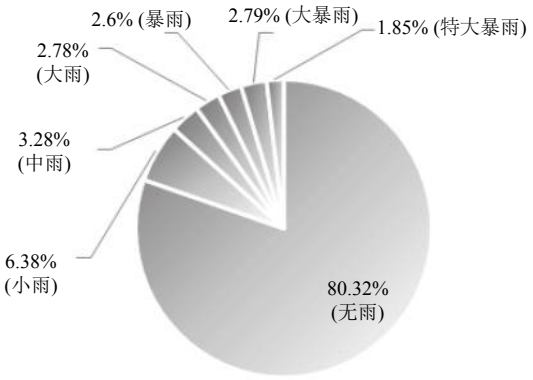

图 3 总体样本降水等级分布图

若 $r>0$, 表示 2 个变量是正相关的; 若 $r<0$, 则表示 2 个变量是负相关, n 为样本数量 (本文选 100 个样本计算相关系数). 其它要素与降雨量之间的相关性初步由 Pearson 求出之后, 还要经过显著性检验再最后判断. 显著性检验公式为:

$$t = \frac{|r_{xy}|}{\sqrt{(1-r_{xy}^2)/(n-2)}} \quad (8)$$

分别计算平均气温、日最低气温、平均水汽压等要素与降雨量的相关系数, 如表 6 所示.

表 6 各要素与降雨量之间的相关系数

序号	气象要素	相关系数
1	平均风速	–0.014 051 753
2	平均气温	0.111 720 713
3	日最低气温	0.184 622 882
4	平均水汽压	0.211 909 646
5	平均相对湿度	0.294 980 431
6	最小相对湿度	0.221 634 957
7	最大风速	–0.052 965 148
8	最大风速的风向	–0.160 114 974
9	极大风速	0.028 312 145
10	极大风速的风向	0.094 995 229
11	日照时数	–0.097 815 752
12	平均本站气压	–0.197 532 962
13	日最高本站气压	–0.214 535 173
14	日最低本站气压	–0.253 301 97
15	小型蒸发量	0.337 908 75
16	大型蒸发量	0.345 026 668

表 6 中平均相对湿度、小型蒸发量及大型蒸发量与降雨量的相关性已通过 0.01 显著性检验, 平均水气压、最小相对湿度、平均本站气压、日最高本站气压、日最低本站气压与降雨量的相关性已通过 0.05 显著性检验, 最后结合气象专家的意见, 选取了平均气温、日最低气温、平均水汽压、平均相对湿度、最小相对湿度、平均风速、最大风速、最大风速的风向、极大风速、极大风速的风向、日照时数这 11 个要素作为预测因子. 网络结构为 11-h-7 三层网络, 其中 11 为输入层的节点数目, 即 11 个预测因子作为网络的输入向量, h 为隐含层的数目, $h=3\sqrt{2}+a$ (a 为 1~10 之间的调节常数), 7 为输出层节点数目, 即 7 个降水等级, 具体实验参数设置参考文献[15].

3.4 实验结果分析

对 13 个站 06 年降水等级预测结果如表 7 所示, 由表 7 可以看出, 本文提出的算法对整体样本降水等级的预测准确度较高, 达到了 82.4%.

表 7 06 年降水等级预测结果

降水等级	样本数	预测准确数
R_0	3678	3229
R_1	279	215
R_2	122	69
R_3	113	59
R_4	84	41
R_5	119	38
R_6	50	10
共计	4445	3661

与传统的 BP 算法对比实验结果如图 4 所示, 从图中可以看出本文提出的遗传神经网络算法对所有降水等级的预测准确率都要优于传统的 BP 神经网络算法, 对无雨的预测准确度最高, 约 87%, 而对特大暴雨的预测准确度最低, 一方面是因为实验样本中无雨的样本数量占总样本数的 80.32%, 远大于其它降水等级的样本数, 所建立的模型更准确, 而特大暴雨样本数量仅占总样本数的 1.85%, 对模型的训练不充分, 另一方面是因为特大暴雨的成因很复杂, 我们实验仅仅选取了众多因子中相关性较高的 11 个因子, 这是远远不够的, 实际情况要复杂的多, 需要结合各种方法综合预测才行.

为了检测所提出方法的扩展性和高效性, 分别在不同的节点数进行了实验, 运行时间结果如图 5 所示, 由图 5 我们不难看出, 随着 DataNode 节点数量的增加, 算法的运行时间明显缩短, 说明本文提出的方法在 Hadoop 平台上有很高的运行效率和扩展性.

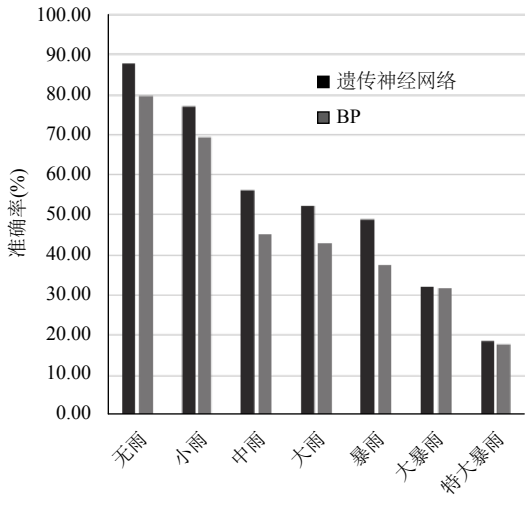

图 4 实验结果对比图

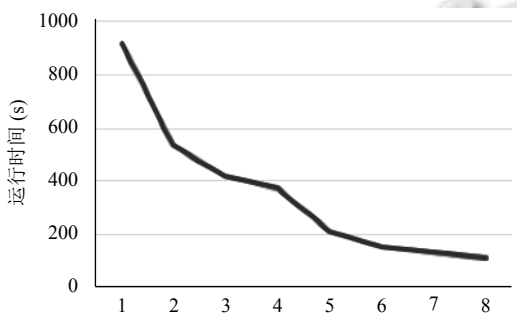

图 5 不同节点运行时间图

4 结论与展望

本文基于 Hadoop 大数据离线分析平台构建了基于遗传神经网络算法的天气预报方法, 并以天津市 13 个台站 1951~2006 年的地面气候资料日值数据为基础, 建立了遗传神经网络预测模型, 最后以降雨量等级为决策属性进行了实验. 结果表明:

1) 本文提出的方法对降水等级的预测准确率优于传统的 BP 算法, 整体预测准确率达到 82.4%, 其中对于等级 R_0 的预测精度最高, 高达 87%.

2) 该方法可以有效的处理海量气象数据, 同时具有很高的运行效率和良好的扩展性, 为天气预报提拱了一种全新的思路和方法.

本文的方法对降水因子仅考虑了相关性比较高的 11 个因子, 对特大暴雨预测准确率很低, 还有一些方面需要进一步研究, 比如可以结合 T213 数值天气预报, 筛选出一些非常规的气象要素来进行降水等级的预测, 以及寻找更有效的算法来解决这个问题, 这些方面都很有意义, 值得在未来的研究中仔细钻研.

参考文献

1 李海涛, 刘云生, 兰长杰. 基于 Hadoop 的生物质能源工程数据资源管理平台. 计算机系统应用, 2018, 27(5): 80–85. [doi: 10.15888/j.cnki.csa.006341]

2 杨淑群, 芮景析, 冯汉中. 支持向量机 (SVM) 方法在降水分类预测中的应用. 西南农业大学学报 (自然科学版), 2006, 28(2): 252–257. [doi: 10.3969/j.issn.1673-9868.2006.02.020]

3 胡邦辉, 刘善亮, 席岩, 等. 一种 Bayes 降水概率预报的最优子集算法. 应用气象学报, 2015, 26(2): 185–192. [doi: 10.11898/1001-7313.20150206]

4 Prasad N, Kumar P, Mm N. An approach to prediction of precipitation using gini index in SLIQ decision tree. Proceedings of the 4th International Conference on Intelligent Systems, Modelling and Simulation. Bangkok, Thailand. 2013. 56–60.

5 王军, 费凯, 程勇. 基于改进的 Adaboost-BP 模型在降水中的预测. 计算机应用, 2017, 37(9): 2689–2693. [doi: 10.11772/j.issn.1001-9081.2017.09.2689]

6 Wu JS, Long J, Liu MZ. Evolving RBF neural networks for rainfall prediction using hybrid particle swarm optimization and genetic algorithm. Neurocomputing, 2015, 148: 136–142. [doi: 10.1016/j.neucom.2012.10.043]

7 胡健伟, 周玉良, 金菊良. BP 神经网络洪水预报模型在洪水预报系统中的应用. 水文, 2015, 35(1): 20–25. [doi: 10.3969/j.issn.1674-9405.2015.01.005]

8 赵正佳, 黄洪钟, 陈新. 优化设计求解的遗传-神经网络新算法研究. 西南交通大学学报, 2000, 35(1): 65–68. [doi: 10.3969/j.issn.0258-2724.2000.01.016]

9 郭强, 朱若函, 张晓萌. 基于遗传禁忌算法优化的模糊神经网络垂直切换算法. 计算机应用研究, 2016, 33(3): 840–842, 847. [doi: 10.3969/j.issn.1001-3695.2016.03.045]

10 谢建宏. 基于并行量子遗传神经网络的自诊断智能结构传感器的优化配置. 计算机应用研究, 2012, 29(3): 919–922. [doi: 10.3969/j.issn.1001-3695.2012.03.033]

11 Jajodia S, Samarati P, Sapino ML, *et al*. Flexible support for multiple access control policies. ACM Transactions on Database Systems, 2001, 26(2): 214–260. [doi: 10.1145/383891.383894]

12 金龙, 吴建生, 林开平, 等. 基于遗传算法的神经网络短期气候预测模型. 高原气象, 2005, 24(6): 981–987. [doi: 10.3321/j.issn:1000-0534.2005.06.019]

13 宋连春, 肖风劲, 李威. 我国现代化气候业务现状及未来发展. 应用气象学报, 2013, 24(5): 513–520. [doi: 10.3969/j.issn.1001-7313.2013.05.001]

14 殷长春, 孙思源, 高秀鹤, 等. 基于局部相关性约束的三维大地电磁数据和重力数据的联合反演. 地球物理学报, 2018, 61(1): 358–367. [doi: 10.6038/cjg2018K0765]

15 陈闯, Chellali R, 邢尹. 改进遗传算法优化 BP 神经网络的语音情感识别. 计算机应用研究, 2019, 36(2): 344–346, 361.